64'er
4. PROGRAMM
SONDERHEFT
Grundlagen und Tips für Commodore 64
Neue Top-Listings zum Abtippen
SONDERHEFT 4/1985
DM 14,-
64'er
Grafik
★ Faszination in 3D
★ Super-Grafik-erweiterungen
★ 80-Zeichenkarte zum Abtippen
Drucker
★ Jede Menge Hardcopy-Routinen
★ Centronics-Interface für 30 Mark
★ Neue Basic-Befehle für Drucker
Grundlagen
★ Sprites ★ Hardcopies
★ Scrolling ★ Schnittstellen
EPSON
FX-80
Alle Programme auch auf Diskette erhältlich

Beeindruckende Grafik

Zwei Themen interessieren den Anwender eines C 64 besonders: Grafik und Drucker. Die Grafik zum einen, weil der C 64 dafür die besten Voraussetzungen bietet, der Drucker zum anderen, da man seine Kunstwerke nicht nur auf dem vergänglichen Bildschirm betrachten möchte, sondern auch etwas »in Händen« halten will.

Deshalb wurden beide Themen in diesem vierten Sonderheft der 64'er zusammengefaßt. Drucker und Grafik haben auch für sich alleine betrachtet ihre Existenzberechtigung, gemeinsam sind sie jedoch unschlagbar.

Was die Grafik anbelangt, finden Sie hier einige Programme, die das Herz höher schlagen lassen. Ein Schwerpunkt liegt in der dreidimensionalen Darstellung von Körpern auch auf einem Heimcomputer. Zwei Programme verdienen dabei Ihr besonderes Augenmerk: »3D-Supergrafik« und »Räumliche Grafik auf dem C 64«.

Erzeugen Sie räumliche Körper, betrachten sie von allen Seiten, lassen das Licht von verschiedenen Richtungen einfallen und geben das Ganze auf einem Drucker aus.

Um »normale« Grafik auf dem C 64 betreiben zu können, bieten wir Ihnen eine Reihe von Basic- und Grafikerweiterungen. Eine mit reiner Software erzeugte 80-Zeichen-Darstellung ist ebenso Bestandteil wie komfortable Befehlserweiterungen für Drucker.

Hardcopy-Routinen für die verschiedensten Drucker mit Leistungsmerkmalen wie vierfache Größe oder Umsetzung der Farben in Graustufen sind in einer eigenen Rubrik zusammengefaßt.

Wer sich seine eigene Hardcopy-Routine basteln will, aber noch keine Ahnung hat wie das geht, erfährt im Abschnitt

Grundlagen, wie man das macht. Außerdem werden die Grundzüge der Spriteprogrammierung und des Fein-Scrolling behandelt.

Alles in allem erhalten Sie wieder rund 28 Listings der Spitzenklasse. Alle Programme wurden mit Prüfsummen versehen, um Ihnen das Abtippen zu erleichtern. Geben Sie daher bitte vorher die beiden Listings »Checksummer 64 V3« und »MSE« in Ihren C 64 ein. Damit können Sie dann alle anderen Listings sicher und bequem abtippen.

(Albert Absmeier)

Listing-Service

Wer keine Zeit oder keine Lust hat, alle Programme selbst in mühevoller Kleinarbeit abzuschreiben, kann wieder auf den bewährten Leserservice zurückgreifen. Alle Listings befinden sich auf einer Diskette mit der Bestellnummer L6'85S4A. Die Diskette kostet 29,90 Mark.

Bitte verwenden Sie für Ihre Bestellung die beigefügte Postscheck-Zahlkarte zur Überweisung des Rechnungsbetrags. Sie erleichtern uns die Auftragsabwicklung und Sie **sparen** sich die **3,— DM** Versandkosten!

Bestell-Nr.: L6 85 S4A / DM 29,90

Vorwort

Prüfsummenlistings

Drucker-Anwendung

Basic-Erweiterung

Hardcopy-Routinen

Zeichengenerator

Grafik

Grundlagen

Tips und Tricks

Rubriken

Checksummer 64 — Neu

Der Checksummer 64 V3 überprüft jede Basic-Zeile direkt nach der Eingabe, erkennt Fehleingaben und auch Vertauschungen von Zahlen und Ziffern, und erspart deshalb eine aufwendige Fehlersuche.

Der Checksummer 64 V3 ist ein kleines Maschinenprogramm, das Sie sofort unterrichtet, ob Sie die jeweilige Programmzeile korrekt eingegeben haben.

So gehen Sie vor:

1. Programm abtippen und speichern.

2. Starten mit RUN

3. Nach kurzer Zeit sehen Sie am Bildschirm: CHECKSUMMER 64, CHECKSUMMER AKTIVIERT, AUSSCHALTEN MIT POKE 1,55, ANSCHALTEN MIT POKE 1,53, READY.

4. Anschalten des Checksummer 64 V3 mit POKE 1,53.

5. Test: Geben Sie in einer freien Zeile ein: »1 REM« und drücken die RETURN-Taste. Am Bildschirm oben links sollten Sie die Prüfsumme <63> sehen.

6. Geben Sie ein Listing aus unserem Heft ein. Nach jeder Zeile wird die Zahl, die im Listing in Klammern < > steht, in den Bildschirm eingeblendet. Stimmen die Zahlen nicht überein, so liegt vermutlich ein Eingabefehler vor. **Die Zahl in den Klammern, und auch die Klammern selbst, dürfen beim Abtippen nicht mit eingegeben werden!**

7. Dieser neue Checksummer 64 V3 bemerkt, im Gegensatz zu den bisherigen, auch Vertauschungen von Zahlen und Buchstaben.

8. Unsere Basic-Listings enthalten keine Steuerzeichen mehr. Diese werden ersetzt durch Klartext und stehen zwischen geschweiften Klammern. Deshalb sind weder die Klammern noch was dazwischen steht, abzutippen, sondern die in Tabelle 1 aufgeführten Tasten zu drücken. Auf Ihrem Bildschirm erhalten Sie dann wieder die entsprechenden Grafikzeichen (siehe Bild 1 und 2).

9. Alle Grafikzeichen werden ebenfalls ersetzt durch unterstrichene oder überstrichene Großbuchstaben. Unterstrichene Buchstaben bedeuten, daß Sie die SHIFT-Taste und den angegebenen Buchstaben drücken müssen, überstrichene jedoch die Commodore-Taste mit dem Buchstaben. Auch hier erhalten Sie am Bildschirm das entsprechende Grafikzeichen und nicht etwa das im Listing erkennbare Zeichen (siehe Bild 1 und 2).

Checksummer VC 20 V3

Der Checksummer VC 20 V3 ist im Prinzip genauso aufgebaut wie der Checksummer 64. Da beim VC 20 jedoch nicht die Möglichkeit besteht, das ROM softwaremäßig zu modifizieren, mußte ein anderer Weg als beim Commodore 64 gewählt werden, um die Checksumme zu generieren.

In ihrer Funktionsweise unterscheiden sich der Checksummer VC 20 und der Checksummer 64 nicht. Es gelten folgende Sonderregelungen bei der Benutzung des Checksummer VC 20:

— Da der Basic-Bereich nicht belegt werden soll, ist das Programm im Kassettenpuffer abgelegt.

— Angeschaltet wird der Checksummer VC 20 mit »SYS 955«.

CTRL steht für Control-Taste, so bedeutet [CTRL-A], daß Sie die Control-Taste und die Taste »A« drücken müssen. Im folgenden steht:

{DOWN}	Taste neben rechtem Shift, Cursor unten
{UP}	Shift-Taste & Taste neben rechtem Shift; Cursor hoch
{CLR}	Shift-Taste & 2. Taste ganz rechts oben
{INST}	Shift-Taste & Taste ganz rechts oben
{HOME}	2. Taste von ganz rechts oben
{DEL}	Taste ganz rechts oben
{RIGHT}	Taste ganz rechts unten
{LEFT}	Shift-Taste & Taste unten rechts
{SPACE}	Leertaste
{F1}	grauer Tastenblock rechts
{F3}	grauer Tastenblock rechts
{F5}	grauer Tastenblock rechts
{F7}	grauer Tastenblock rechts
{F2}	grauer Tastenblock rechts & Shift
{F4}	grauer Tastenblock rechts & Shift
{F6}	grauer Tastenblock rechts & Shift
{F8}	grauer Tastenblock rechts & Shift
{RETURN}	Shift-Taste & Return
{BLACK}	Control-Taste & 1
{WHITE}	Control-Taste & 2
{RED}	Control-Taste & 3
{CYAN}	Control-Taste & 4
{PURPLE}	Control-Taste & 5
{GREEN}	Control-Taste & 6
{BLUE}	Control-Taste & 7
{YELLOW}	Control-Taste & 8
{RVSON}	Control-Taste & 9
{RVOFF}	Control-Taste & 0
{ORANGE}	Commodore-Taste & 1
{BROWN}	Commodore-Taste & 2
{LIG.RED}	Commodore-Taste & 3
{GREY 1}	Commodore-Taste & 4
{GREY 2}	Commodore-Taste & 5
{LIG.GREEN}	Commodore-Taste & 6
{LIG.BLUE}	Commodore-Taste & 7
{GREY 3}	Commodore-Taste & 8

Wenn Sie sich erst einmal an die in Klartext geschriebenen Steuerzeichen gewöhnt haben, werden Sie den Vorteil dieser Schreibweise erkennen. Der zu dem jeweiligen Steuerzeichen gehörende Klartext ist so verfaßt, daß Sie leicht die Taste beziehungsweise die Tastenkombination finden, die Sie drücken müssen.

Die Steuerbefehle im Klartext

— Abschaltung des Checksummer VC 20 wird mit »SYS 58459« vollzogen.

Achtung: Nehmen Sie keine Kassetten-Operationen vor, wenn der Checksummer VC 20 eingeschaltet ist. Da das Betriebssystem den Kassettenpuffer mit Daten belegt, kann der Checksummer VC 20 überschrieben werden, was zur Folge hat, daß sich der Computer bei aktiviertem Checksummer VC 20 »aufhängt«. Wollen Sie deshalb ein Programm auf (von) Kassette abspeichern (laden), so müssen Sie erst den Checksummer VC 20 abschalten (SYS 58459).

Daraufhin kann der Kassettenpuffer mit Daten überschrieben werden, ohne daß der Computer »aussteigt«.

Als Sicherung wird bei der Initialisierung geprüft, ob das zuletzt angesprochene Peripherie-Gerät der Kassettenrecorder war. Ist das der Fall, so werden die Betriebssystemroutinen LOAD und SAVE für die Benutzung gesperrt. Der Rechner meldet bei Aufruf einer dieser beiden Routinen READY, ohne weitere Aktionen durchzuführen. Diese Sicherung kann man nach der Tipparbeit aufheben, wenn man den Checksummer VC 20 mit SYS 58459 abschaltet. Dadurch wird der Kassettenpuffer für andere Daten freigemacht. Weiterhin wird dann durch gleichzeitiges Drücken der Tasten »Run-Stop & Restore« erreicht, daß die Betriebssystemroutinen LOAD und SAVE wieder eingerichtet werden.

— Bei Benutzung einer Diskettenstation brauchen Sie nicht darauf zu achten, daß bei LOAD beziehungsweise SAVE der

Checksummer VC 20 überschrieben wird, da der Kassettenpuffer für die Diskettenstation normalerweise nicht genutzt wird. Deshalb können Sie die beiden Routinen weiterhin normal nutzen, sofern der Rechner bei der Initialisierung des Checksummer VC 20 feststellt, daß das zuletzt angesprochene Peripherie-Gerät nicht der Kassettenrecorder war.

— Bedingt durch den anderen Aufbau des Checksummer VC 20 wird anders als beim Checksummer 64 nach der LOAD-Routine keine Checksumme ausgegeben.

— Wird eine Zeile gelöscht, also eine Zahl zwischen 0 und 65999 eingegeben, und danach Return gedrückt, so wird eine Checksumme ausgegeben, die aber keine Bedeutung hat.

Sie können die Programme auch weiterhin ohne den Checksummer eintippen. (F. Lonczewski/gk)

Hinweis: {13 SPACE} bedeutet 13mal die Leertaste drücken

```
1 REM ******************************           <139>
2 REM *                            *           <051>
3 REM *      CHECKSUMMER 64 V3     *           <153>
4 REM *                            *           <053>
5 REM *    WRITTEN MAERZ 1985 BY   *           <210>
6 REM *                            *           <055>
7 REM *      FRANK LONCZEWSKI      *           <039>
8 REM *                            *           <057>
9 REM ******************************           <147>
10 PRINT"{CLR,11SPACE,RVSON}CHECKSUMMER 64
   V3{RVOFF}"                                  <194>
11 PRINT"{2DOWN,9SPACE}EINEN MOMENT, BITTE
   ..."                                        <130>
12 FOR I=828 TO 864:READ A:POKE I,A:PS=PS+
   A+1:NEXT I                                  <018>
13 IF PS<>5802 THEN PRINT"PRUEFSUMMENFEHLE
   R IN ZEILEN 20-22":END                      <100>
14 SYS 828:PS=0:FOR I=58464 TO 58583:READ
   A:POKE I,A:PS=PS+A+1:NEXT I                 <084>
15 IF PS<>16267 THEN PRINT"PRUEFSUMMENFEHL
   ER IN ZEILEN 22-30":END                     <193>
16 POKE 1,53:POKE 42289,96:POKE 42290,228      <150>
17 PRINT"{4DOWN,9SPACE}CHECKSUMMER AKTIVIE
   RT."                                        <107>
18 PRINT"{2DOWN}AUSSCHALTEN : POKE1,55"        <180>
19 PRINT"{DOWN}ANSCHALTEN{2SPACE}: POKE1,5
   3":NEW                                      <185>
20 DATA 169,0,133,254,162,1,189,93,3,133,2
   55,160,0,177,254                            <089>
21 DATA 145,254,136,208,249,230,255,165,25
   5,221,95,3,208,238,202                      <042>
22 DATA 16,230,96,160,224,192,0,160,2,169,
   0,170,133,254,177                           <084>
23 DATA 95,240,40,201,32,208,3,200,208,245
   ,133,255,138,41,7                           <249>
24 DATA 170,240,14,72,165,255,24,42,105,0,
   202,208,249,133,255                         <078>
25 DATA 104,170,232,165,255,24,101,254,133
   ,254,76,111,228,192,4                       <005>
26 DATA 48,219,198,214,165,214,72,162,3,16
   9,32,157,1,4,189                            <177>
27 DATA 212,228,32,210,255,208,12,0,92,72,
   32,201,255,170,104                          <065>
28 DATA 144,1,138,96,202,16,228,166,254,16
   9,0,32,205,189,169                          <125>
29 DATA 62,32,210,255,104,133,214,32,108,2
   29,169,141,32,210,255                       <088>
30 DATA 76,128,164,9,60,18,19                  <034>
```

© 64'er

Dieser neue Checksummer 64 V3 erkennt auch Vertauschungen von Zahlen.

```
10 REM***************                          <154>
11 REM*              *                         <060>
12 REM* CHECKSUMMER *                          <159>
13 REM*              *                         <062>
14 REM* V3      VC20 *                         <216>
15 REM*              *                         <064>
16 REM*   WRITTEN    *                         <066>
17 REM* MAERZ  1985 *                          <015>
18 REM*      BY      *                         <122>
19 REM*F. LONCZEWSKI*                          <107>
20 REM***************                          <164>
21 PRINT"{CLR,SPACE,RVSON}CHECKSUMMER V3 V
   C-20{RVOFF}"                                <079>
22 PRINT"{2DOWN}EINEN MOMENT, BITTE..."        <141>
23 FOR I=827 TO 1019:READ A:POKE I,A           <109>
24 PS=PS+A+1:NEXT I                            <205>
25 IF PS<>24464 THEN PRINT"{DOWN}PRUEFSUMM
   ENFEHLER !":END                             <080>
26 SYS 981:PRINT"CHECKSUMMER AKTIVIERT."       <083>
27 PRINT"AN :SYS981"                           <161>
28 PRINT"{DOWN}AUS:SYS58459, BEI CAS-{4SPA
   CE}SETTE ZUSAETZLICH{5SPACE}RUN/STOP &
   RESTORE"                                    <229>
29 PRINT"{DOWN}BEI AKTIVIERTEM CHECK-SUMME
   R KEIN";                                    <069>
30 PRINT" CASSETTEN-BETRIEB (LOAD, SAVE){2
   SPACE}ERLAUBT!":NEW                         <023>
31 DATA 32,95,3,134,122,132,123,32,115,0,1
   70,240,243,162,255                          <192>
32 DATA 134,58,144,10,162,0,134,255,32,121
   ,197,76,225,199,162                         <039>
33 DATA 1,134,255,76,156,196,166,255,224,1
   ,240,3,76,96,197                            <185>
34 DATA 160,2,169,0,170,133,254,177,95,240
   ,40,201,32,208,3                            <143>
35 DATA 200,208,245,133,253,138,41,7,170,2
   40,14,72,165,253,24                         <249>
36 DATA 42,105,0,202,208,249,133,253,104,1
   70,232,165,253,24,101                       <040>
37 DATA 254,133,254,76,119,3,192,4,48,219,
   198,214,165,214,72                          <199>
38 DATA 162,3,169,32,157,1,4,189,209,3,32,
   210,255,202,16                              <253>
39 DATA 242,166,254,169,0,32,205,221,169,6
   2,32,210,255,104,133                        <240>
40 DATA 214,32,135,229,169,141,32,210,255,
   162,0,134,255,240,148                       <235>
41 DATA 9,60,18,19,169,59,141,2,3,169,3,14
   1,3,3,165                                   <084>
42 DATA 186,201,1,208,16,169,116,141,48,3,
   141,50,3,169,196                            <197>
43 DATA 141,49,3,141,51,3,173,136,2,141,17
   0,3,96                                      <079>
```

© 64'er

Der neue Checksummer VC 20 V3 erkennt auch Vertauschungen von Zahlen.

```
5 PRINT CHR$(14)                               <242>
10 PRINT"{CLR}"                                <254>
20 PRINT"Q****************************W"       <130>
30 PRINT"{4DOWN,2SPACE}TEST{SPACE,BLUE,6SP
   ACE}"                                       <022>
40 PRINT"BBBBBBBBBBBBBBBBBBBBBBBBBBBB"         <108>
```

© 64'er

Bild 1. So könnte ein Teil eines Listings abgedruckt sein. In Zeile 10 müssen Sie nach den Anführungsstrichen die CLEAR/HOME-Taste drücken und nicht die Klammern mit dem Wort CLR. In Zeile 20 drücken Sie nach den Anführungsstrichen die Commodore-Taste und den Buchstaben Q, gefolgt von mehreren SHIFT und Stern-Taste, und zum Schluß die Commodore-Taste und den Buchstaben W. In Zeile 30 ist es viermal die Cursor-nach-unten-Taste, gefolgt von zweimal die Leertaste, dann SHIFT und T und normal EST, zum Schluß noch einmal die Leertaste, die Farbtaste Blau (Control und 7) und sechsmal die Leertaste. Zeile 40 besteht lediglich aus mehreren Grafikzeichen, die mit der Commodore-Taste und B erzeugt werden.

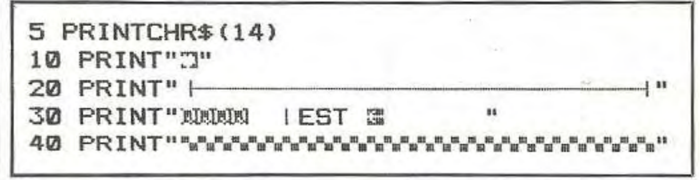


Bild 2. Auf dem Bildschirm oder Ihrem Drucker sieht das Listing (Bild 1) so aus.

MSE – Abtippen sicher und leicht gemacht

Ähnlich wie der »Checksummer« ist auch der MSE ein Hilfsmittel bei der Eingabe von Listings, diesmal jedoch bei reinen Maschinensprache-Programmen.

Im Gegensatz zum »Checksummer« aber ist die Eingabe nicht ohne den MSE möglich. Der MSE verringert die Tipparbeit um ein Drittel und schließt Fehleingaben vollkommen aus. Außerdem können Sie die Werte blind eingeben, ohne andauernd auf den Bildschirm schauen zu müssen. Dies wird durch akustische Meldungen realisiert.

MSE ist ein Maschinenspracheditor, mit dem ein Vertippen ausgeschlossen ist. Eine abgetippte Zeile wird nur angenommen, wenn sie richtig ist. Eine Checksumme am Ende jeder Zeile prüft, ob die richtigen Werte in der richtigen Zeile an der richtigen Stelle stehen. Wenn nicht, ertönt ein Warnsignal, und man beseitigt den Fehler.

War die Zeile korrekt, erklingt ein Gong, und die nächste Zeilennummer wird ausgegeben. Damit ist also auch »blindes« Eintippen möglich; Sie können sich voll auf den Text konzentrieren.

So arbeitet man mit MSE

Laden und starten Sie MSE. Zuerst wird der Programmname und die Start- und Endadresse erfragt. **Diese Angaben entnehmen Sie dem Kopf des jeweiligen abgedruckten Listings.** MSE meldet sich dann mit der Zeilennummer der ersten Zeile. Wenn Sie die Zeile richtig eingegeben haben, erscheint die nächste Zeilennummer und so weiter bis zum Ende. Zum Schluß wird das fertige Programm mit »CTRL-S« auf Diskette oder Kassette abgespeichert. Dazu sind keine weiteren Angaben mehr erforderlich. Das Programm kann dann ganz normal wieder geladen und gestartet werden. Wenn Sie nicht alles auf einmal tippen wollen, können Sie jederzeit unterbrechen und den eingetippten Teil mit »CTRL-S« abspeichern. Wollen Sie weiterarbeiten, laden und starten Sie MSE wieder.

Geben Sie auf die Frage nach der Startadresse aber jetzt »CTRL-L« ein, um Ihr Teilprogramm zu laden. Jetzt können Sie mit »CTRL-N« die Adresse eingeben, an der Sie weitertippen müssen. Wenn Sie sich nicht gemerkt haben, wie weit Sie gekommen sind, geben Sie nach dem Laden »CTRL-M« ein.

Auf die Frage nach der Startadresse antworten Sie mit der Anfangsadresse, die links in der Kopfzeile auf dem Bildschirm steht. Nun wird Ihr Programm aufgelistet. Mit »SPACE« wird das Listen fortgesetzt, mit »STOP« abgebrochen. Das Ende Ihres Programmteils erkennen Sie sehr einfach daran, daß nur noch der Wert »AA« in der Zeile steht. Die Adresse dieser Zeile müssen Sie anschließend mit »CTRL-N« eingeben. Das Programm ist nur mit »STOP/RESTORE« zu verlassen. Speichern Sie aber vorher unbedingt immer Ihren Text ab.

Hinweise zum Abtippen

Vor dem Abtippen oder späteren Wiederladen des MSE-**Laders** müssen Sie unbedingt folgende Zeile eingeben:

POKE 43,1: POKE 44,32: POKE 8192,0: NEW

Starten Sie das Programm mit RUN. Fehlerhafte Zeilen werden angezeigt und müssen korrigiert werden, bis der Lader zum »READY« durchläuft. Jetzt müssen Sie das fertige MSE-Programm abspeichern. Dazu brauchen Sie nur »RETURN« zu drücken, weil die erforderlichen Angaben schon auf dem Bildschirm stehen. (Kassettenbesitzer müssen in Zeile 343 die letzte Zahl in »1« abändern.) Ab jetzt können Sie »MSE V1.0« direkt, also ohne den DATA-Lader, benutzen. MSE V1.0 wird ganz normal mit »,8« geladen (keine POKEs notwendig).

(N. Mann / D. Weineck / gk)

MSE-Befehle:

DEL	löscht die letzte Eingabe.
CTRL-S	speichert das eingetippte Programm ab.
CTRL-L	lädt ein Programm. Start- und Endadresse werden automatisch ermittelt.
CTRL-M	listet den Speicherinhalt. Abbruch mit STOP-Taste, weiter mit Leertaste.
CTRL-N	erlaubt die Eingabe einer neuen Adresse zum Weitertippen.
CTRL-P	gibt ein MSE-Listing auf dem Drucker aus.

```
100 REM ******************************       <091>
110 REM *                            *       <159>
120 REM *         M S E   LADER      *       <206>
130 REM *                            *       <179>
220 REM ******************************       <211>
230 REM                                      <036>
240 DIM H(75): FOR I=0 TO 9                  <113>
250 H(48+I)=I: H(65+I)=I+10:NEXT             <041>
260 FOR I=2048 TO 3755 : READ A$             <198>
270 H=ASC(LEFT$(A$,1)):L=ASC(RIGHT$(A$,1))   <199>
280 D=H(H)*16+H(L):S=S+D:POKE I,D            <219>
290 A=A+1:IF A<20 THEN NEXT:A=-1             <141>
300 PRINT " ZEILE:";1000+Z;                  <011>
310 READ V :Z=Z+1:IF V=S THEN 330            <218>
320 PRINT"PRUEFSUMMENFEHLER !":STOP          <138>
330 IF A<0 THEN 341                          <221>
340 S=0:A=0:PRINT:NEXT                       <046>
341 PRINT"{CLR}PO43,1:PO44,8:PO45,172:PO46
    ,14                                      <010>
342 POKE 631,19:POKE 632,13:POKE 633,13:PO
    KE 198,3                                 <249>
343 PRINT"{3DOWN}SAVE"CHR$(34)"MSE V1.0"CH
    R$(34)",8                                <171>
344 END                                      <092>
```

```
350 REM :                                    <111>
360 REM       ******************             <149>
370 REM       *      DATA      *             <078>
380 REM       ******************             <169>
390 REM :                                    <151>
1000 DATA 00,0B,08,0A,00,9E,32,30,36,31,00
     ,00,00,A2,08,A9,36,85,A4,A9, 1247       <119>
1001 DATA 08,85,A5,A9,00,85,A6,A9,B0,85,A7
     ,A0,00,B1,A4,91,A6,C8,D0,F9, 2888       <054>
1002 DATA E6,A5,E6,A7,CA,D0,F2,A9,36,85,01
     ,4C,00,B0,20,D1,B1,A9,06,8D, 2787       <144>
1003 DATA 21,D0,A9,03,8D,20,D0,8D,86,02,A0
     ,B3,A9,74,20,FF,B1,A0,B3,A9, 2667       <237>
1004 DATA B9,20,FF,B1,A0,00,20,CF,FF,99,01
     ,02,C8,C9,0D,D0,F5,88,F0,D2, 2912       <217>
1005 DATA C0,0F,90,02,A0,0E,8C,00,02,20,EA
     ,B1,A0,B3,A9,CF,20,FF,B1,20, 2323       <013>
1006 DATA 8E,B4,85,FC,85,62,20,8E,B4,85,FB
     ,85,61,20,A7,B4,D0,20,A0,B3, 2864       <199>
1007 DATA A9,E5,20,FF,B1,20,8E,B4,85,60,20
     ,8E,B4,85,5F,20,A7,B4,D0,0A, 2624       <091>
1008 DATA A5,61,C5,5F,A5,62,E5,60,90,06,20
     ,43,B3,4C,3A,B0,A9,AA,A0,00, 2379       <167>
1009 DATA 91,FB,E6,FB,D0,02,E6,FC,20,3F,B2
```

Der MSE zum bequemen Abtippen von Assemblerprogrammen. Beachten Sie den Checksummer auf Seite 6.

```
     ,90,EF,4C,FB,B4,A2,02,86,58, 3118      <152>
1010 DATA A9,A6,A0,9D,20,F2,B1,20,E4,FF,F0
     ,FB,C9,30,90,0C,C9,47,B0,08, 2970      <231>
1011 DATA C9,3A,90,0B,C9,41,B0,07,C9,14,D0
     ,0F,4C,0B,B1,20,D2,FF,A6,58, 2322      <121>
1012 DATA 95,F7,C6,58,D0,D2,60,AE,8D,02,F0
     ,26,C9,0C,D0,03,4C,0B,B6,C9, 2685      <057>
1013 DATA 13,D0,03,4C,8B,B5,C9,0D,D0,03,4C
     ,BA,B4,C9,10,D0,03,4C,68,B5, 2282      <225>
1014 DATA C9,0E,D0,06,20,5F,B4,4C,64,B1,4C
     ,92,B0,A5,F9,20,02,B1,0A,0A, 2132      <208>
1015 DATA 0A,0A,85,F9,A5,F8,20,02,B1,05,F9
     ,60,C9,3A,90,02,69,08,29,0F, 1950      <092>
1016 DATA 60,A6,59,E0,08,90,1F,A6,58,E0,02
     ,B0,06,20,D2,FF,4C,8E,B0,C6, 2509      <188>
1017 DATA 59,A0,14,A9,92,20,F2,B1,CA,D0,FA
     ,84,57,68,68,4C,8B,B1,A6,D3, 2891      <197>
1018 DATA E0,08,B0,03,4C,92,B0,20,D2,FF,A6
     ,58,E0,02,90,09,C6,59,20,D2, 2468      <049>
1019 DATA FF,C6,58,D0,F9,4C,8E,B0,48,4A,4A
     ,4A,4A,20,59,B1,68,29,0F,C9, 2419      <035>
1020 DATA 0A,90,02,69,06,69,30,4C,D2,FF,A2
     ,FC,9A,20,D1,B1,20,48,B2,20, 2261      <073>
1021 DATA EA,B1,20,9F,B2,A5,FC,20,4E,B1,A5
     ,FB,20,4E,B1,20,ED,B1,A9,3A, 2860      <148>
1022 DATA A0,20,20,F2,B1,A9,00,85,59,20,8E
     ,B0,20,ED,B1,A4,59,20,EF,B0, 2530      <233>
1023 DATA 91,FB,C8,84,59,C0,08,90,EC,20,10
     ,B2,A9,12,20,D2,FF,20,8E,B0, 2657      <105>
1024 DATA 20,EF,B0,C5,FF,F0,0D,20,43,B3,A9
     ,14,A0,14,20,F2,B1,4C,A2,B1, 2665      <034>
1025 DATA A9,92,20,D2,FF,20,33,B2,20,E0,B2
     ,20,3F,B2,90,9F,4C,8B,B5,A9, 2648      <123>
1026 DATA 93,20,D2,FF,A2,00,A9,03,9D,00,D8
     ,9D,00,D9,9D,00,DA,9D,00,DB, 2476      <237>
1027 DATA E8,D0,EF,60,A9,0D,2C,A9,20,4C,D2
     ,FF,20,D2,FF,98,4C,D2,FF,20, 2965      <160>
1028 DATA E4,FF,F0,FB,60,84,5D,85,5C,A0,00
     ,B1,5C,F0,06,20,D2,FF,C8,D0, 3100      <077>
1029 DATA F6,60,A5,FB,85,5A,A0,00,84,5B,B1
     ,FB,18,65,5A,85,5A,90,02,E6, 2606      <156>
1030 DATA 5B,06,5A,26,5B,C8,C0,08,90,EC,A5
     ,5A,65,5B,85,FF,60,18,A5,FB, 2467      <219>
1031 DATA 69,08,85,FB,90,02,E6,FC,60,A5,FB
     ,C5,5F,A5,FC,E5,60,60,A0,B3, 3106      <183>
1032 DATA A9,FB,20,FF,B1,A0,01,B9,00,02,20
     ,D2,FF,CC,00,02,C8,90,F4,A9, 2692      <098>
1033 DATA 10,ED,00,02,AA,20,ED,B1,CA,D0,FA
     ,A5,62,20,4E,B1,A5,61,20,4E, 2453      <236>
1034 DATA B1,20,ED,B1,A5,60,20,4E,B1,A5,5F
     ,20,4E,B1,A9,9F,20,D2,FF,20, 2575      <038>
1035 DATA EA,B1,24,5E,10,01,60,A9,12,20,D2
     ,FF,A2,28,20,ED,B1,CA,D0,FA, 2646      <161>
1036 DATA A9,92,4C,D2,FF,A5,D6,C9,16,B0,01
     ,60,A9,A0,85,A4,A9,78,85,A6, 2945      <204>
1037 DATA A9,04,85,A5,85,A7,A2,13,A0,27,B1
     ,A4,91,A6,88,10,F9,CA,F0,19, 2671      <208>
1038 DATA 18,A5,A4,69,28,85,A4,90,02,E6,A5
     ,18,A5,A6,69,28,85,A6,90,E0, 2503      <251>
1039 DATA E6,A7,4C,B6,B2,A9,91,4C,D2,FF,A9
     ,0F,8D,18,D4,A9,00,8D,05,D4, 2776      <000>
1040 DATA A9,F7,8D,06,D4,A9,11,8D,04,D4,A9
     ,32,8D,01,D4,A9,00,8D,00,D4, 2413      <126>
1041 DATA A0,80,20,09,B3,A9,10,8D,04,D4,60
     ,A2,FF,CA,D0,FD,88,D0,F8,60, 2914      <240>
1042 DATA A9,0F,8D,18,D4,A9,2D,8D,05,D4,A9
     ,A5,8D,06,D4,A9,21,8D,04,D4, 2385      <119>
1043 DATA A9,07,8D,01,D4,A9,05,8D,00,D4,A0
     ,FF,20,09,B3,A9,20,8D,04,D4, 2250      <078>
1044 DATA A9,00,8D,01,D4,8D,00,D4,60,38,20
     ,F0,FF,8A,48,98,48,18,A0,06, 2179      <175>
1045 DATA A2,18,20,F0,FF,A0,B4,A9,0A,20,FF
     ,B1,20,12,B3,20,E4,FF,F0,FB, 2931      <093>
1046 DATA A2,1D,A9,14,20,D2,FF,CA,D0,FA,68
     ,A8,68,AA,18,4C,F0,FF,0D,0D, 2704      <088>
1047 DATA 0D,20,20,20,20,20,20,20,4D,41,53
     ,43,48,49,4E,45,4E,53,50,52, 1144      <216>
1048 DATA 41,43,48,45,20,2D,20,45,44,49,54
     ,4F,52,20,0D,0D,20,20,20,20, 1023      <038>
1049 DATA 20,20,20,20,56,4F,4E,20,4E,2E,4D
     ,41,4E,4E,20,26,20,44,2E,57, 1128      <206>
1050 DATA 45,49,4E,45,43,4B,00,0D,0D,0D,20
     ,20,20,50,52,4F,47,52,41,4D, 1102      <117>
1051 DATA 4D,4E,41,4D,45,20,3A,20,00,0D,0D
     ,20,20,20,53,54,41,52,54,41, 1073      <095>
1052 DATA 44,52,45,53,53,45,20,3A,20,24,00
     ,0D,0D,20,20,20,45,4E,44,41, 1014      <129>
1053 DATA 44,52,45,53,53,45,20,20,20,3A,20
     ,24,00,92,05,20,50,52,4F,47, 1171      <217>
1054 DATA 52,41,4D,4D,20,3A,20,00,12,20,20
     ,2A,2A,2A,20,46,41,4C,53,43, 1024      <027>
1055 DATA 48,45,20,45,49,4E,47,41,42,45,20
     ,2A,2A,2A,20,20,92,00,0D,0D, 1058      <098>
1056 DATA 2A,2A,2A,20,45,4E,44,45,20,2A,2A
     ,2A,00,13,05,20,20,12,44,92, 920       <148>
1057 DATA 49,53,4B,20,4F,44,45,52,20,12,54
     ,92,41,50,45,0D,00,13,20,20, 1151      <035>
1058 DATA 49,2F,4F,20,2D,20,46,45,48,4C,45
     ,52,00,20,D1,B1,20,48,B2,A0, 1606      <012>
1059 DATA B3,A9,CF,20,FF,B1,20,8E,B4,85,FC
     ,20,8E,B4,85,FB,C5,61,A5,FC, 3207      <251>
1060 DATA E5,62,90,23,A5,FB,C5,5F,A5,FC,E5
     ,60,B0,19,20,A7,B4,D0,14,60, 2860      <112>
1061 DATA 20,A7,B4,F0,0C,85,F9,20,A7,B4,F0
     ,05,85,F8,4C,EF,B0,68,68,20, 2749      <088>
1062 DATA 43,B3,4C,5F,B4,20,CF,FF,C9,4C,D0
     ,09,20,D1,B1,20,48,B2,4C,0B, 2372      <046>
1063 DATA B6,C9,0D,60,A9,00,85,5E,20,5F,B4
     ,20,EA,B1,20,0D,B5,24,5E,30, 2042      <120>
1064 DATA 05,20,E4,FF,F0,FB,20,E1,FF,F0,26
     ,20,9F,B2,24,5E,10,09,20,4E, 2435      <198>
1065 DATA B5,20,0D,B5,20,60,B5,20,33,B2,20
     ,3F,B2,90,D7,A0,B4,A9,28,20, 2190      <207>
1066 DATA FF,B1,20,E4,FF,C9,0D,D0,F9,A9,00
     ,85,5E,A5,61,85,FB,A5,62,85, 3056      <240>
1067 DATA FC,20,E0,B2,4C,64,B1,A5,FC,20,4E
     ,B1,A5,FB,85,FF,20,4E,B1,A9, 3003      <221>
1068 DATA 20,A0,3A,20,F2,B1,A0,00,20,ED,B1
     ,B1,FB,20,4E,B1,C8,C0,08,90, 2566      <070>
1069 DATA F3,20,ED,B1,24,5E,30,03,A9,12,2C
     ,A9,20,20,D2,FF,20,10,B2,A5, 2190      <059>
1070 DATA FF,20,4E,B1,A9,92,20,D2,FF,4C,EA
     ,[illegible],A9,FF,85,B8,85,B9,A9,04, 3073      <029>
1071 DATA 85,BA,20,C0,FF,A2,FF,4C,C9,FF,20
     ,CC,FF,A9,FF,4C,C3,FF,20,5F, 3315      <189>
1072 DATA B4,A9,80,85,5E,20,4E,B5,20,48,B2
     ,A2,24,A9,2D,20,D2,FF,CA,D0, 2596      <111>
1073 DATA FA,20,EA,B1,20,EA,B1,20,60,B5,4C
     ,C1,B4,20,B8,B5,A6,5F,A4,60, 2812      <015>
1074 DATA A9,61,20,D8,FF,B0,0A,20,B7,FF,29
     ,BF,D0,03,4C,FB,B4,A9,01,20, 2577      <201>
1075 DATA C3,FF,20,68,B6,A0,B4,A9,4F,20,FF
     ,B1,20,F9,B1,4C,FB,B4,20,68, 2921      <237>
1076 DATA B6,A9,37,A0,B4,20,FF,B1,20,F9,B1
     ,A2,08,C9,44,F0,06,A2,01,C9, 2717      <213>
1077 DATA 54,D0,F1,A9,01,A8,20,BA,FF,A0,00
     ,E0,01,F0,1A,A9,40,8D,20,02, 2403      <101>
1078 DATA A9,3A,8D,21,02,B9,01,02,99,22,02
     ,C8,CC,00,02,90,F4,C8,C8,D0, 2182      <127>
1079 DATA 0C,B9,01,02,99,20,02,C8,CC,00,02
     ,D0,F4,98,A2,20,A0,02,4C,BD, 2018      <025>
1080 DATA FF,20,B8,B5,A5,BA,C9,08,90,33,A6
     ,B9,86,57,A9,01,20,C3,FF,A9, 2800      <022>
1081 DATA 60,85,B9,20,C0,FF,B0,28,A5,BA,20
     ,B4,FF,A5,B9,20,96,FF,20,A5, 2911      <053>
1082 DATA FF,85,61,A5,90,4A,4A,B0,13,20,A5
     ,FF,85,62,20,AB,FF,A5,57,85, 2663      <214>
1083 DATA B9,A9,00,20,D5,FF,90,03,4C,A3,B5
     ,86,5F,84,60,A5,BA,C9,01,D0, 2639      <131>
1084 DATA 0A,AD,3D,03,85,61,AD,3E,03,85,62
     ,4C,FB,B4,A9,13,20,D2,FF,A2, 2300      <120>
1085 DATA 1C,20,ED,B1,CA,D0,FA,60, 1230     <214>
```

© 64'er

MSE (Schluß). Dieses Listing können Sie (müssen aber nicht) mit dem neuen Checksummer 64 V3 in diesem Heft eingeben.

Centronics-Schnittstelle für Seikosha GP-550A

Mit diesem Software-Interface können Sie auf einem Seikosha GP-550A die Steuer- und Grafikzeichen des C 64 listen und drucken. Der Wert der Schnittstelle wird dabei durch eine Hardcopy-Funktion noch um ein Vielfaches erhöht.

Der Seikosha-Drucker GP-550A besitzt in der Grundversion eine Centronics-Schnittstelle. Zum Anschluß an den C 64 ist deshalb ein Interface notwendig. Unser Leser Pius Nippgen wollte sich aber die hohen Kosten für ein Hardware-Interface sparen. Er dachte sich, was mit einem Epson funktioniert, muß auch mit dem GP-550A gehen. Er nahm sich dazu kurzerhand die Eyssele-Schnittstelle aus Ausgabe 7/84 (auch in diesem Sonderheft) vor und schrieb sie auf den GP-550A um. Diese Schnittstelle besteht aus einem Programmteil und einem Kabel vom User-Port des C 64 an den Centronics-Eingang des Druckers.

Die gesamte Ansteuerung (Sekundär- und Geräteadressen) wurde beibehalten. Die Adressen und die Verbindungsvorschrift des Kabels können Sie bei der Eyssele-Schnittstelle nachschlagen.

Der GP-550A wird mit anderen Befehlssequenzen angesteuert als ein Epson-Drucker. Diese Sequenzen sind nötig, um beispielsweise den Grafikmodus des Druckers einzuschalten. Deshalb war die erste Aufgabe, alle Steuer-Codes der Eyssele-Schnittstelle dem GP-550A entsprechend zu ändern. Aber nicht nur das. Ein Epson-Drucker ordnet, beginnend mit der untersten der acht Nadeln, steigende binäre Wertigkeiten zu, der GP-550A aber fallende:

Epson	Nadel	Gp-550A
128	8	1
64	7	2
32	6	4
16	5	8
8	4	16
4	3	32
2	2	64
1	1	128

Zusätzlich zu diesen Änderungen wurde eine Hardcopy-Funktion (Ausdruck des Grafikspeichers) in die Eyssele-Schnittstelle aufgenommen. Die Hardcopy-Routine wird nach dem Öffnen des Druckerkanals, beispielsweise mit OPEN 19,19 mit SYS 52223 aufgerufen. Der Aufruf kann sowohl innerhalb eines Programms, als auch im Direktmodus erfolgen. Es können alle Grafikseiten gedruckt werden, auch solche im RAM unterhalb des ROMs.

Mit ein paar POKE-Befehlen läßt sich die Hardcopy variieren. Durch POKE 52236,224 wird der Speicherbereich von \$E000 bis \$FFFF selektiert. 224 ist das Highbyte von \$E000 (57344). Erfolgt kein POKE-Befehl, wird vom Programm der normale Grafikbereich von \$2000 (8192) bis \$3FFF (16383) angenommen. Verschiedene Hardcopies können stufenlos aneinander gedruckt werden, indem man einfach mehrmals nacheinander mit SYS 52223 die Routine aufruft. Vorher muß man nur mit POKE 52236,X den gewünschten Grafikbereich einstellen.

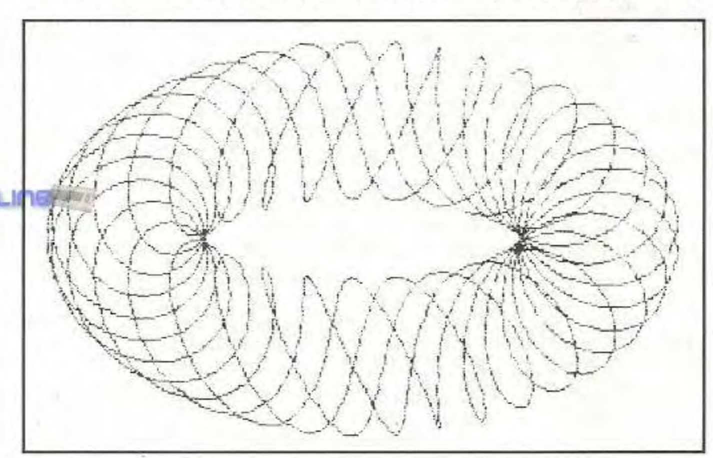
Flexible Hardcopy

Über die Speicherzelle 52242 läßt sich die Zeilenzahl der Hardcopy einstellen. POKE 52242,25 bedeutet, daß die gesamte Grafikseite (25 Zeilen) ausgedruckt wird.

Einen invertierten Ausdruck erreichen Sie durch POKE 52296,165. Durch POKE52310,0:POKE52239,50:POKE 52340,51 wird die Hardcopy-Breite auf die Hälfte reduziert, das Bild also gestaucht.

Initialisiert wird die Schnittstelle mit SYS 51456. Sowohl nach dem Laden als auch nach RUN/STOP-RESTORE.

Geben Sie das Listing mit dem MSE ein. Danach können Sie mit LOAD "name",8,1 oder 1,1 laden. Nach dem Laden sollten Sie noch NEW eingeben, damit die Basic-Vektoren wieder vernünftig gesetzt werden. Ohne NEW erhalten Sie bei der nächsten Anweisung einen »Out of Memory Error«. Wie Sie das Druckerkabel bauen müssen, entnehmen Sie bitte dem Artikel über die Eyssele-Schnittstelle. (Pius Nippgen/hm)



Hardcopy mit dem Seikosha GP-550A

```
930:   cbd2 1b               .byte27      ;grafik-steuersequenz
932:   cbd3 47               .byte71      ;fuer
934:   cbd4 30               .bvte"0"     :orafikzeichen-ausoabe
936:   cbd5 30               .byte"0"     ;seikosha gp 550 a
938:   cbd6 38               .byte"8"
                      ;                          ;
                      ;--------------------------;
                      ;  'umdreh'-routine        ;
                      ; der bit-wetigkeiten      ;
                      ;der grafikzeichebbytes    ;
                      ;--------------------------;
                      ;                          ;
954:   cbd7 a9 80            lda  #$80       ;bit7-wertigkeit
956:   cbd9 85 97            sta  $97
958:   cbdb a9 00            lda  #$00
960:   cbdd a8               tay
962:   cbde 85 fb            sta  $fb
964:   cbe0 bd c0 02         lda  $02c0,x    ;das zu 'drehende'
966:   cbe3 85 fc            sta  $fc        ;byte laden
                      ;
970:   cbe5 a5 fc     loop   lda  $fc
972:   cbe7 25 97            and  $97        ;pruefen, ob
974:   cbe9 f0 07            beq  $cbf2      ;bit gesetzt
                      ;
978:   cbeb a5 fb            lda  $fb        ;byte-zwischenspeicher
980:   cbed 19 77 cc         ora  $cc77,y
                      ;wenn das jeweilige bit gesetzt ist,
                      ;dann dafuer das entsprechende 'umkehr-bit'
                      ;aus bit-wertigkeits-tabelle holen
988:   cbf0 85 fb            sta  $fb
990:   cbf2 c8               iny
                      ;bit-tabellen-zeiger erhoehen
                      ;
996:   cbf3 18               clc
998:   cbf4 46 97            lsr  $97
                      ;naechst-niedrigeres bit pruefen
1002:  cbf6 b0 02            bcs  $cbfa
                      ;wenn alle bits geprueft
```

```
1006:  cbf8 90 eb              bcc  loop
                        ;wenn noch nicht alle bits geprueft
1010:  cbfa a5 fb              lda  $fb
                        ;fertig 'gedrehtes' byte laden
1014:  cbfc 4c 20 cb           jmp  $cb20
                        ;und in eyssele-routine zurueckspringen
                        ;                    ;
                        ;--------------------;
                        ; hardcopy - routine ;
                        ;--------------------;
                        ;                    ;
1028:  cbff a9 1b              lda  #$1b     ;steuersequenz, welche
1030:  cc01 20 40 c9           jsr  $c940    ;passenden zeilenabstand
1032:  cc04 a9 39              lda  #$39     ;fuer grafikausdruck
1034:  cc06 20 40 c9           jsr  $c940    ;(hardcopy) festlegt
                        ;
1038:  cc09 a9 00              lda  #$00     ;zeiger auf gewuenschten
1040:  cc0b a0 20              ldy  #$20     ;grafikspeicher
1042:  cc0d 85 fd              sta  $fd      ;setzen
1044:  cc0f 84 fe              sty  $fe
                        ;
1048:  cc11 a2 19              ldx  #$19
                        ;anzahl der auszudruckenden
                        ;hardcopyzeilen steht in x-register
1054:  cc13 a0 04       zeile  ldy  #$04     ;grafiksteuersequenz fuer
1056:  cc15 b9 72 cc           lda  $cc72,y ;eine hardcopyzeile
1058:  cc18 20 40 c9           jsr  $c940    ;an drucker ueber
1060:  cc1b 88                 dey           ;userport senden
1062:  cc1c 10 f7              bpl  $cc15
1064:  cc1e a9 28              lda  #$28     ;40 zeichen
1066:  cc20 85 15              sta  $15      ;pro hardcopyzeile
                        ;
1070:  cc22 a9 80       loop1  lda  #$80     ;bit7-wertigkeit
1072:  cc24 85 97              sta  $97
                        ;speicher des jeweiligen pruef-bits
                        ;
1078:  cc26 a9 00       loop2  lda  #$00
1080:  cc28 85 14              sta  $14
                        ;zwischenspeicher des spaltenbytes
1084:  cc2a a0 07              ldy  #$07
                        ;in 8 zeilenbytes pruefen, ob jeweiliges
                        ;pruef-bit ($97) gesetzt ist
                        ;
1092:  cc2c a9 34              lda  #$34     ;ganzen ram einschalten
1094:  cc2e 78                 sei           ;indem in speicherstelle1
1096:  cc2f 85 01              sta  $01      ;bit 0,1 geloescht
                        ;
                        ;umformung der zeilenbytes des grafikspeichers
                        ;in spaltenbytes fuer drucker
1104:  cc31 b1 fd       loop3  lda  ($fd),y
                        ;zeilenbyte lesen
1108:  cc33 25 97              and  $97
                        ;in $97 steht jeweils das bit,auf welches
                        ;die 8 zeilenbytes geprueft werden
1114:  cc35 f0 07              beq  $cc3e
1116:  cc37 a5 14              lda  $14
1118:  cc39 19 77 cc           ora  $cc77,y
1120:  cc3c 85 14              sta  $14
                        ;entsprechendes bit aus tabelle holen
                        ;und im spaltenbyte-zwischenspeicher setzen
1126:  cc3e 88                 dey
                        ;jeweiliges bit in allen 8 zeilenbytes
                        ;pruefen, ob gesetzt
1132:  cc3f 10 f0              bpl  loop3
1134:  cc41 a9 02              lda  #$02
1136:  cc43 85 fb              sta  $fb
                        ;'zaehler' fuer zweimalige ausgabe
1140:  cc45 a9 ff              lda  #$ff
1142:  cc47 38                 sec
1144:  cc48 e5 14              sbc  $14
                        ;spaltenbyte (=byte fuer drucker)
                        ;invertieren
1150:  cc4a a0 37              ldy  #$37     ;rom wieder
1152:  cc4c 84 01              sty  $01      ;einschalten
1154:  cc4e 58                 cli
1156:  cc4f 20 40 c9           jsr  $c940
                        ;spaltenbyte an drucker
1160:  cc52 18                 clc
1162:  cc53 46 fb              lsr  $fb
1164:  cc55 90 ee              bcc  $cc45
                        ;zweimal spaltenbyte an drucker ausgeben,
                        ;wodurch hardcopy doppelt so breit wird
1170:  cc57 46 97              lsr  $97
                        ;naechst-niedrigeres bit
                        ;der 8 zeilenbytes pruefen
1176:  cc59 90 cb              bcc  loop2
1178:  cc5b a5 fd              lda  $fd      ;grafikspeicherzeiger
1180:  cc5d 69 07              adc  #$07     ;um 8 bytes erhoehen
1182:  cc5f 85 fd              sta  $fd      ;erforderlichenfalls
1184:  cc61 90 02              bcc  $cc65    ;highbyte des
1186:  cc63 e6 fe              inc  $fe      ;zeigers erhoehen
1188:  cc65 c6 15              dec  $15      ;pruefen, ob
1190:  cc67 d0 b9              bne  loop1    ;hardcopyzeile fertig.
1192:  cc69 a9 0a              lda  #$0a     ;line-feed an
1194:  cc6b 20 40 c9           jsr  $c940    ;drucker senden
1196:  cc6e ca                 dex
1198:  cc6f d0 a2              bne  zeile
                        ;pruefen, ob alle hardcopyzeilen
                        ;ausgegeben sind, also ob hardcopy
                        ;fertig ist
1206:  cc71 60                 rts
                        ;ende der hardcopy-routine,
                        ;ruecksprung in programm
                        ;bzw. direktmodus
                        ;
1216:  cc72 30                 .byte"0"      ;grafiksteuersequenz
1218:  cc73 34                 .byte"4"      ;fuer eine
1220:  cc74 36                 .byte"6"      ;hardcopyzeile
1222:  cc75 47                 .byte71
1224:  cc76 1b                 .byte27

1228:  cc77 01                 .byte1        ;bit-wertigkeits-
1230:  cc78 02                 .byte2        ;tabelle
1232:  cc79 04                 .byte4
1234:  cc7a 08                 .byte8
1236:  cc7b 10                 .byte16
1238:  cc7c 20                 .byte32
1240:  cc7d 40                 .byte64
1242:  cc7e 80                 .byte128
```

Bild 1. Die wichtigsten Teile des Centronics-Interface als Source-Code

```
programm : gp550a              c900 cc80
-----------------------------------------

c900 : a9 5a a0 c9 8d 1a 03 8c   06
c908 : 1b 03 a9 91 a0 c9 8d 1c   08
c910 : 03 8c 1d 03 a9 ad a0 c9   1f
c918 : 8d 1e 03 8c 1f 03 a9 c8   49
c920 : a0 c9 8d 20 03 8c 21 03   2b
c928 : a9 e3 a0 c9 8d 26 03 8c   53
c930 : 27 03 a9 ff 8d 03 dd ad   07
c938 : 02 dd 09 04 8d 02 dd 60   0d
c940 : 48 a9 10 2c 0d dd f0 fb   62
c948 : 68 8d 01 dd ad 00 dd 09   d7
c950 : 04 8d 00 dd 29 fb 8d 00   7f
c958 : dd 60 a6 b8 f0 05 20 0f   fc
c960 : f3 d0 03 4c fe f6 a6 98   79
c968 : e0 0a 90 03 4c fb f6 e6   20
c970 : 98 a5 b8 9d 59 02 a5 b9   6c
c978 : 09 60 9d 6d 02 a5 ba 9d   3a
c980 : 63 02 c9 04 f0 04 c9 10   4e
c988 : 90 02 18 60 c9 00 4c 77   e8
c990 : f3 20 14 f3 f0 02 18 60   57
c998 : 20 1f f3 8a 48 a5 ba c9   c6
c9a0 : 10 b0 07 c9 04 f0 03 4c   70
c9a8 : 9d f2 4c f1 f2 20 0f f3   64
c9b0 : f0 03 4c 01 f7 20 1f f3   3a
c9b8 : a5 ba c9 04 f0 04 c9 10   24
c9c0 : 90 03 4c 0a f7 4c 19 f2   52
c9c8 : 20 0f f3 f0 03 4c 01 f7   11
c9d0 : 20 1f f3 a5 ba c9 04 f0   1d
c9d8 : 04 c9 10 90 03 4c 75 f2   25
c9e0 : 4c 5b f2 48 85 9e a5 9a   b9
c9e8 : c9 10 b0 07 c9 04 f0 03   4d
c9f0 : 4c cd f1 98 48 8a 48 a5   f7
c9f8 : 9e a4 9a c0 10 d0 06 20   87
ca00 : 40 c9 18 90 1f c0 11 d0   1b
ca08 : 06 20 5c ca 18 90 15 c0   6b
ca10 : 04 f0 04 c0 12 d0 06 20   a5
ca18 : 2b ca 18 90 07 c0 13 d0   25
ca20 : 03 20 44 ca 68 aa 68 a8   6d
ca28 : 68 18 60 c9 41 90 12 c9   62
ca30 : 5f b0 04 09 20 d0 0a c9   4e
ca38 : c1 90 06 c9 de b0 02 29   ca
ca40 : 7f 4c 40 c9 c9 ff f0 18   bf
ca48 : c9 60 b0 03 4c 40 c9 e9   90
ca50 : 40 10 02 e9 40 a0 d0 84   ab
ca58 : 06 4c c4 ca c9 ff d0 06   fb
ca60 : a2 5e a0 d0 d0 5b 48 a4   c6
ca68 : b9 c0 ff d0 02 e6 b9 29   2c
ca70 : 7f c9 20 90 2c a8 a5 b9   00
ca78 : 29 01 f0 10 68 c9 a0 90   d8
ca80 : 04 c9 c0 90 03 4c 2b ca   80
ca88 : e9 40 d0 0e 68 c9 60 b0   3f
ca90 : 03 4c 40 c9 e9 40 10 02   e7
ca98 : e9 40 a0 d0 84 06 4c c4   17
caa0 : ca 68 24 0f 30 03 4c 40   56
caa8 : c9 18 69 40 30 02 69 40   19
cab0 : aa a5 b9 29 02 d0 71 a0   6e
cab8 : d0 a5 b9 29 01 f0 02 a0   cf
```

Listing 1. Das Centronics-Interface für den Seikosha GP-550A. Beachten Sie die Hinweise zum Abtippen auf Seite 8.

```
cac0 : d8 84 06 8a a8 a9 00 a2   ca
cac8 : 07 9d c0 02 ca 10 fa 98   58
cad0 : 4a 4a 4a 4a 4a 18 65 06   22
cad8 : 85 06 98 0a 0a 0a 85 05   d9
cae0 : a9 01 85 03 78 a5 01 29   d7
cae8 : fb 85 01 a0 07 b1 05 85   17
caf0 : 02 a2 07 06 02 90 08 bd   06
caf8 : c0 02 05 03 9d c0 02 ca   d9
cb00 : 10 f1 06 03 88 10 e6 a5   db
cb08 : 01 09 04 85 01 58 a2 00   9d
cb10 : bd d2 cb 20 40 c9 e8 e0   e5
cb18 : 05 d0 f5 a2 07 4c d7 cb   21
cb20 : 20 40 c9 ca 10 f7 18 60   0e
cb28 : e0 d5 90 01 ca 8a 29 7f   dc
cb30 : c9 50 90 02 e9 03 c9 45   ee
cb38 : 90 02 e9 03 c9 41 90 02   91
cb40 : e9 21 c9 1c 90 02 e9 08   81
cb48 : c9 11 90 02 e9 0b 38 e9   aa
cb50 : 04 aa a9 3c 20 40 c9 a0   08
cb58 : ff ca f0 08 c8 b9 7a cb   d5
cb60 : 10 fa 30 f5 c8 b9 7a cb   94
cb68 : 30 06 20 40 c9 18 90 f4   35
cb70 : 29 7f 20 40 c9 a9 3e 4c   e4
cb78 : 40 c9 57 48 d4 43 52 c4   b6
cb80 : 52 4f ce 48 4f cd 52 45   6e
cb88 : c4 43 52 d2 47 52 ce 42   a3
cb90 : 4c d5 4f 52 c7 46 b1 46   e7
cb98 : b3 46 b5 46 b7 46 b2 46   a9
cba0 : b4 46 b6 46 b8 42 4c cb   54
cba8 : 43 52 d5 52 4f c6 43 4c   a5
cbb0 : d2 42 52 ce 48 52 d4 47   0b
cbb8 : 52 b1 47 52 b2 48 47 ce   27
cbc0 : 48 42 cc 47 52 b3 50 55   f4
cbc8 : d2 43 52 cc 59 45 cc 43   e3
cbd0 : 59 ce 1b 47 30 30 38 a9   f9
cbd8 : 80 85 97 a9 00 a8 85 fb   89
cbe0 : bd c0 02 85 fc a5 fc 25   6a
cbe8 : 97 f0 07 a5 fb 19 77 cc   6e
cbf0 : 85 fb c8 18 46 97 b0 02   90
cbf8 : 90 eb a5 fb 4c 20 cb a9   af
cc00 : 1b 20 40 c9 a9 39 20 40   da
cc08 : c9 a9 00 a0 20 85 fd 84   e9
cc10 : fe a2 19 a0 04 b9 72 cc   2b
cc18 : 20 40 c9 88 10 f7 a9 28   93
cc20 : 85 15 a9 80 85 97 a9 00   66
cc28 : 85 14 a0 07 a9 34 78 85   e9
cc30 : 01 b1 fd 25 97 f0 07 a5   96
cc38 : 14 19 77 cc 85 14 88 10   8b
cc40 : f0 a9 02 85 fb a9 ff 38   b4
cc48 : e5 14 a0 37 84 01 58 20   38
cc50 : 40 c9 18 46 fb 90 ee 46   d0
cc58 : 97 90 cb a5 fd 69 07 85   31
cc60 : fd 90 02 e6 fe c6 15 d0   1f
cc68 : b9 a9 0a 20 40 c9 ca d0   9c
cc70 : a2 60 30 34 36 47 1b 01   e1
cc78 : 02 04 08 10 20 40 80 00   87
```

Centronics-Schnittstelle für GP-550A (Schluß)

160 Spalten mit dem 1526 oder MPS 802

Dieses Unterprogramm läßt den Commodore 1526-Drucker 128 oder sogar 160 Zeichen pro Zeile drucken. Der Druck von breiten Tabellen wird zum Vergnügen!

Der 1526 kommt durch dieses Programm an die Leistungen von teuren Druckern mit Kleinschriftmodus nahe heran. Mit 160 Zeichen pro Zeile übertrifft er diese Drucker sogar.

Wie Sie wissen ist der 1526 (MPS 802) nur bedingt grafikfähig. Es lassen sich selbstdefinierte Zeichen ausdrucken. Das macht auch dieses Programm. Es setzt zwei zu druckende Buchstaben in ein selbstdefiniertes Zeichen um, das der Drucker zu Papier bringt. Der 1526 druckt also immer zwei Buchstaben gleichzeitig (160 Zeichen pro Zeile). Im 128-Zeichenmodus ist es etwas komplizierter. Hier werden aus acht Buchstaben fünf Druckerzeichen zusammengesetzt (Bild 1). Das Aussehen der Zeichen wird über die DATA-Werte im Programm bestimmt. Acht DATAs für ein Zeichen. Ein DATA-Wert bestimmt, welche Nadeln einer vertikalen Punktreihe angeschlagen werden.

Die einzelnen Nadeln haben die folgenden Wertigkeiten:

```
oben   . 128
       . 64
       . 32
       . 16
       . 8
       . 4
       . 2
unten  . 1
```

Ein Zeichen ist aus acht solchen Punktreihen zusammengesetzt. Die binären Wertigkeiten der einzelnen Punktspalten werden über die CHR$()-Funktion in einem String addiert und an den 1526 gesandt, der das selbstdefinierte Zeichen druckt.

Das folgende kleine Programm druckt einen senkrechten Strich:

```
10 open 5,4,5
20 a$=chr$(255) : rem eine Punktreihe
30 for i=1 to 7 : rem Rest des Zeichens
40 a$=a$+chr$(0): rem mit Leerraum
50 next i :rem auffüllen
60 print#5,a$
70 close5
```

Tips zur Benutzung

Das eigentliche Programm beginnt ab Zeile 60000: Der Teil vorher dient nur zur Demonstration der 128- und der 160-Zeichendarstellung. Ab Zeile 60000 zerlegt das Programm den zu druckenden String (XX$) in seine Einzelteile, baut daraus die selbstdefinierten Zeichen zusammen und schickt diese an den Drucker.

Wenn Sie das Unterprogramm in einem Basic-Programm

```
!"#$%&'()*+,-./0123456789:;<=>?@ABCDEFGHIJKLMNOPQRSTUVWXYZ[\]DAS IST DER ZEICHENSATZ DES PRG'S.
!"#$%&'()*+,-./0123456789:;<=>?@ABCDEFGHIJKLMNOPQRSTUVWXYZ[\]↑
UND DAS IST ZUM VERGLEICH DIE NORMALE GROESSE.DA SIEHT MAN DEN UNTERSCHIED!
```

Bild 1. Die drei Schriftmodi des 1526, die mit diesem Unterprogramm möglich sind

verwenden, übergeben Sie einfach die zu druckenden Zeichen in der Variablen XX$ an das Unterprogramm. Das Unterprogramm wird, wie üblich, mit einem GOSUB gestartet. GOSUB 60000 bewirkt eine 160-Zeichendarstellung, GOSUB 60810 eine »Druckbreite« von 128 Zeichen.

Beachten Sie bitte, daß vor dem Aufruf des 128-Zeichenmodus erst einmal der 160-Zeichenmodus aktiviert werden muß. Warum? Nun, beim ersten Durchlauf des Programmes werden alle DATAs in Variablenfelder (V1$() und V2$()) eingelesen. Da der READ-Befehl grundsätzlich mit dem ersten Datum im ganzen Programm beginnt, müssen zur 128-Zeichendarstellung die 160-Zeichen-DATAs überlesen werden. Das geschieht mit einem GOSUB 60000. Danach sind beide Variablenfelder definiert. Sie können dann beliebig den 128- oder 160-Zeichenmodus aufrufen. Würde zuerst der 128-Zeichenmodus aufgerufen, würden die falschen DATAs gelesen und den Variablen zugeordnet. Ein reichlich seltsames Aussehen der Zeichen wäre die Folge.

Eine detaillierte Programmbeschreibung finden Sie in Tabelle 1. (Martin Riethmüller/hm)

Zeile	Inhalt
0-8	: Programmkopf
18-120	: Demo
59000-	: Unterprogrammkopf 160-Zeichen-Karte
60000	: Eröffnen eines Kanals zum Drucker
60005	: Damit die DATA-Zeilen beim zweiten Aufruf nicht noch einmal aufgerufen werden.
60010-	: DATA-Zeilen für den neuen Zeichensatz
60080	: Dimensionierung der Zeichensatzvariablen und der Druckvariablen, falls nicht schon ab 61000 gemacht
60085	: <SPACE> wird definiert
60090- 60115	: Einlesen der restlichen Buchstaben aus den DATA- : Zeilen und Zuordnung in V1$(II)
60130- 60160	: In 60140 String in einzelne Buchstaben zerlegen. : Dann in 60145 ermitteln der Nummer des Buchstabens, abfangen von nicht vorhandenen Zeichen. In 60155 auszudruckendes neues Zeichen = die letzten zwei Zeichen, gelesen aus dem neudefinierten Zeichensatz in V1$(II).
60165	: Zeilenvorschub des Druckers auf Null
60170- 60185	: Ausdruckschleife mit allmählicher Verschiebung des : Druckkopfes durch LE$
60190	: Zeilenvorschub
60200	: Rückkehr zum Hauptprogramm
60800-	: Unterprogrammkopf 128-Zeichen-Karte
61000	: Eröffnen eines Kanals zum Drucker
61005	: Damit die DATA-Zeilen beim zweiten Aufruf nicht noch einmal aufgerufen werden
61010- 61015	: Dimensionierung der Zeichensatzvariablen und der : Druckvariablen, falls nicht schon ab 61000 gemacht
61020-	: <SPACE> wird definiert
61035-	: DATA-Zeilen, in denen der neue Zeichensatz steckt
61115- 61130	: Einlesen der restlichen Buchstaben aus den DATA- : Zeilen und Zuordnung in V1$(II,JJ)
61140	: Zeilenvorschub des Druckers auf Null
61145- 61220	: In 61155 Zerlegung des eingegebenen Strings in ein- : zelne Buchstaben. Dann in 61160 ermitteln der Nummer des Buchstabens, abfangen von nicht vorhandenen Zeichen. Ab 61170 Zusammenbau von 5 neudefinierten Zeichen aus 8 Zeichen von XX$. (Immer 5 »Streifen« zu einem neuen Zeichen.)
61225- 61235	: Ausdruckschleife mit allmählicher Verschiebung des : Druckkopfes durch LE$.
61245	: Zeilenvorschub
61250	: Rückkehr zum Hauptprogramm

Tabelle 1. Die einzelnen Programmzeilen aufgeschlüsselt

```
0 REM " ******************************         <170>
1 REM " {3SPACE}160/128-ZEICHEN-KARTE{3SP
  ACE}                                        <135>
2 REM " {4SPACE}FUER COMMODORE 1526{4SPAC
  E}                                          <138>
3 REM " {2SPACE}VON MARTIN RIETHMUELLER{2
  SPACE}                                      <080>
4 REM " {2SPACE}BEI FRAGEN ZUM PROGRAMM{2
  SPACE}                                      <095>
5 REM " {3SPACE}ODER ZUM 1526-DRUCKER{3SP
  ACE}                                        <142>
6 REM " GEBE ICH GERNE RAT+AUSKUNFT           <081>
7 REM " {6SPACE}TEL. 07152/47418{5SPACE}      <166>
8 REM " ******************************         <144>
18 CLR:OPEN 1,4                               <153>
19 PRINT"{CLR}"                               <007>
20 REM ***** DEMO *****                       <086>
30 XX$="":FOR X=32 TO 94:XX$=XX$+" "+CHR$(
   X):NEXT                                    <025>
40 XX$=XX$+"DAS IST DER ZEICHENSATZ DES PR
   GMS."                                      <103>
60 GOSUB 60000                                <080>
70 XX$="":FOR X=32 TO 94:XX$=XX$+" "+CHR$(
   X):NEXT                                    <065>
90 GOSUB 61000                                <126>
100 PRINT#1,"UND DAS IST ZUM VERGLEICH DIE
     NORMALE GROESSE.DA SIEHT MAN DEN ";      <080>
110 PRINT#1,"UNTERSCHIED!":GOTO 20            <150>
120 REM ********** DEMO-ENDE ***********      <007>
59000 REM "*********************************
      ************************************
                                              <046>
59010 REM "{7SPACE}UNTERPROGRAMM 160-ZEIC
      HEN-KARTE FUER DRUCKER 1526{10SPACE}
                                              <239>
59020 REM "*********************************
      ************************************
                                              <115>
59030 REM " AUFRUF: GOSUB 60000.DER INHAL
      T DER VARIABLEN XX$ WIRD AUSGEDRUCKT
                                              <002>
59040 REM "*********************************
      ************************************
                                              <135>
59050 REM "{18SPACE}VERWENDETE VARIABLEN{
      28SPACE}                                <147>
59060 REM "{66SPACE}                          <104>
59070 REM " STATUSVARIABLE [ TS ] (ZUR VE
      RMEIDUNG DER MEHRFACHEN EINLESUNG{3S
      PACE}                                   <083>
59080 REM " DER DATAZEILEN IN [ V1$(94) ]
      {36SPACE}                               <065>
59090 REM " LAUFVARIABLEN [ II ] UND [ JJ
       ]{34SPACE}                             <161>
59100 REM " DATALESE-VARIABLE [ AA ]{41SP
      ACE}                                    <220>
59110 REM " ARBEITSVARIABLE [ V1$ ]{42SPA
      CE}                                     <088>
59115 REM " ARBEITSFELD [ W1(2) ]{44SPACE
      }                                       <095>
59120 REM " [ LE$ ] ZUR POSITIONIERUNG DE
      S DRUCKKOPFES{23SPACE}                  <133>
59130 REM " DAS FELD [ V1$(94) ],DAS DEN
      NEUEN ZEICHENSATZ BEINHALTET.{7SPACE
      }                                       <094>
59140 REM " DAS FELD [ DR$(160) ],DAS AUS
      GEDRUCKT WIRD.{22SPACE}                 <029>
59150 REM "*********************************
      ************************************
                                              <028>
60000 OPEN 4,4                                <156>
60005 TS=TS+1:IF TS>1 THEN 60130              <248>
60010 DATA 0,232,0,0,192,0,192,0,112,160,1
      12,160,72,248,176,0,152,32,200,0,80     <081>
60015 DATA 168,80,8,0,64,128,0,112,136,0,0
      ,0,136,112,0,80,32,80,0,32,112,32,0,
      0,4                                     <201>
60020 DATA 24,0,32,32,32,0,0,0,8,0,24,32,1
      92,0                                    <048>
60025 DATA 248,136,248,0,64,192,248,0,184,
      168,232,0,168,168,248,0,96,184,32       <159>
60030 DATA 0,232,168,184,0,248,168,184,0,1
      28,160,248,0,248,168,248,0,232,168,2
      48                                      <001>
60035 DATA 0,0,40,0,0,0,4,40,0,32,80,80,13
      6,80,80,80,0,136,80,80,32,128,168,22
```

Listing 1. 128/160 Zeichen auf dem 1526 (MPS 802). Beachten Sie bitte zur Eingabe den Checksummer 64.

```
       4                                          <100>
60040 DATA 0,112,168,200,080                      <121>
60045 DATA 120,160,120,0,248,168,112,0,112
      ,136,136,0,248,136,112,0,248,168,168
      ,0                                          <201>
60050 DATA 248,160,128,0,248,136,152,0,248
      ,32,248,0,0,248,0,0,136,136,240,0           <215>
60055 DATA 248,112,136,0,248,8,8,0                <127>
60060 DATA 248,64,192,248,248,64,56,0,112,
      136,112,0,248,160,224,0,112,152,120,
      0                                           <021>
60065 DATA 248,176,232,0,104,168,176,0,128
      ,248,128,0,248,8,248,0,240,8,240,0,2
      48                                          <106>
60070 DATA 8,56,248,216,32,216,0,224,40,24
      8,0,152,168,200,0                           <143>
60077 DATA 0,248,136,0,208,168,72,0,0,136,
      248,0,64,248,64,0                           <246>
60080 DIM V1$(94):IF TZ<1 THEN DIM DR$(160
      )                                           <243>
60085 V1$(32)=CHR$(0)+CHR$(0)+CHR$(0)+CHR$
      (0)                                         <201>
60090 FOR II=33 TO 94                             <020>
60095 FOR JJ=1 TO 4                               <184>
60100 READ AA                                     <213>
60105 V1$(II)=V1$(II)+CHR$(AA):AA=0               <249>
60110 NEXT JJ                                     <163>
60115 NEXT II                                     <144>
60130 FOR II=1 TO 80                              <039>
60135 FOR JJ=1 TO 2                               <222>
60140 V1$=MID$(XX$,II*2-(2-JJ),1):IF V1$="
      "THEN V1$=" "                               <072>
60145 W1(JJ)=ASC(V1$):IF ASC(V1$)>94 OR AS
      C(V1$)<32 THEN W1(JJ)=32                    <211>
60150 NEXT JJ                                     <203>
60155 DR$(80+II)=V1$(W1(1))+V1$(W1(2))            <081>
60160 NEXT II                                     <191>
60165 OPEN 6,4,6:PRINT#6,CHR$(0):CLOSE 6          <145>
60170 OPEN 5,4,5                                  <245>
60175 FOR II=1 TO 80                              <086>
60180 PRINT#5,DR$(80+II):PRINT#4,LE$+CHR$(
      254)                                        <193>
60185 LE$=LE$+" ":NEXT II                         <147>
60190 CLOSE 5:LE$="":OPEN 6,4,6:PRINT#6,CH
      R$(23):PRINT#4,:PRINT#6,CHR$(32):CLO
      SE 6                                        <164>
60200 XX$="":CLOSE 4:RETURN                       <087>
60800 REM "A*********************************
      *************************************
      S                                           <068>
60810 REM "={7SPACE}UNTERPROGRAMM 128-ZEIC
      HEN-KARTE FUER DRUCKER 1526{10SPACE}
      =                                           <101>
60820 REM "G*********************************
      *************************************
      W                                           <137>
60830 REM "= AUFRUF MIT SYS61000.DER INHAL
      T DER VARIABLEN XX$ WIRD AUSGEDRUCKT
      =                                           <137>
60840 REM "G*********************************
      *************************************
      W                                           <157>
60850 REM "={18SPACE}VERWENDETE VARIABLEN{
      28SPACE}=                                   <169>
60860 REM "={66SPACE}=                            <126>
60870 REM "= STATUSVARIABLE [ TZ ] (ZUR VE
      RMEIDUNG DER MEHRFACHEN EINLESUNG{3S
      PACE}=                                      <074>
60880 REM "= DER DATAZEILEN IN [ V2$(94,5)
       ]{34SPACE}=                                <220>
60890 REM "= LAUFVARIABLEN [ II ] UND [ JJ
       ]{34SPACE}=                                <183>
60900 REM "= DATALESE-VARIABLE [ AA ]{41SP
      ACE}=                                       <242>
60910 REM "= ARBEITSVARIABLE [ V2$ ]{42SPA
      CE}=                                        <174>
60915 REM "= ARBEITSFELD [ F(8) ]{45SPACE}
      =                                           <214>
60920 REM "= [ LE$ ] ZUR POSITIONIERUNG DE
      S DRUCKKOPFES{23SPACE}=                     <155>
60930 REM "= DAS FELD [ V2$(94,5) ],DAS DE
      N NEUEN ZEICHENSATZ BEINHALTET.{5SPA
      CE}=                                        <026>
60940 REM "= DAS FELD [ DR$(160) ],DAS AUS
      GEDRUCKT WIRD.{22SPACE}=                    <051>
60950 REM "Z*********************************
      *************************************
      X                                           <050>
61000 OPEN 4,4:OPEN 5,4,5                         <250>
61005 TZ=TZ+1:IF TZ>1 THEN 61140                  <092>
61010 DIM V2$(94,5)                               <241>
61015 IF TS<1 THEN DIM DR$(180)                   <061>
61020 FOR II=1 TO 5                               <070>
61025 V2$(32,II)=CHR$(0)                          <238>
61030 NEXT II                                     <043>
61035 DATA 0,0,250,0,0,0,224,0,224,0,20,62
      ,20,62,20,98,159,242,140,0                  <030>
61040 DATA 198,200,16,38,198,108,154,150,1
      10,9,0,0,96,128,0,124,130,130,0,0           <199>
61045 DATA 130,130,124,0,0,40,16,124,16,40
      ,16,16,124,16,16,0,0,1,6,0                  <075>
61050 DATA 16,16,16,16,0,0,0,6,6,0,6,8,16,
      32,192,124,138,146,124,0                    <238>
61055 DATA 32,64,254,0,0,70,138,146,98,0,1
      46,146,146,108,0,24,40,94,8,0               <004>
61060 DATA 242,146,146,140,0,60,82,146,140
      ,0,144,158,176,208,0,108,146,146,108
      ,0                                          <111>
61065 DATA 100,146,146,124,0,0,0,18,0,0,0,
      1,12,0,0,16,40,68,68,0                      <190>
61070 DATA 40,40,40,40,0,68,68,40,16,0,64,
      128,154,96,0,124,130,178,116,0              <033>
61075 DATA 254,144,144,254,0,254,146,146,1
      08,0,124,130,130,130,0,254,130,130,1
      24                                          <052>
61080 DATA 0,254,146,146,146,0,254,144,144
      ,128,0,124,130,146,92,0,254,16,16,25
      4,0                                         <089>
61085 DATA 0,0,190,0,0,132,130,130,252,0,2
      54,16,40,198,0,254,2,2,2,0                  <241>
61090 DATA 254,64,32,64,254,254,64,32,254,
      0,124,130,130,124,0,254,144,144,96,0        <023>
61095 DATA 124,130,142,126,1,254,152,148,9
      8,0,98,146,146,140,0,128,254,128,128
      ,0                                          <106>
61100 DATA 252,2,2,2,252,0,248,6,6,248,0,254
      ,2,4,2,254,198,56,56,198,0                  <084>
61105 DATA 224,18,18,252,0,134,154,178,210
      ,0,254,130,130,0,0,76,178,146,82,0          <090>
61110 DATA 0,130,130,254,0,32,64,254,64,32        <163>
61115 FOR II=33 TO 94                             <029>
61120 FOR JJ=1 TO 5                               <194>
61125 READ AA:V2$(II,JJ)=CHR$(AA)                 <011>
61130 NEXT JJ:NEXT II                             <123>
61140 OPEN 6,4,6:PRINT#6,CHR$(0):CLOSE 6          <102>
61145 FOR II=1 TO 16                              <043>
61150 FOR JJ=1 TO 8                               <227>
61155 V2$=MID$(XX$,II*8-(8-JJ),1):IF V2$="
      "THEN V2$=" "                               <154>
61160 F(JJ)=ASC(V2$):IF ASC(V2$)<32 OR ASC
      (V2$)>94 THEN F(JJ)=32                      <199>
61165 NEXT JJ                                     <202>
61170 DR$(II*5-4)=V2$(F(1),1)+V2$(F(1),2)+
      V2$(F(1),3)+V2$(F(1),4)+V2$(F(1),5)         <059>
61175 DR$(II*5-4)=DR$(II*5-4)+V2$(F(2),1)+
      V2$(F(2),2)+V2$(F(2),3)                     <037>
61180 DR$(II*5-3)=V2$(F(2),4)+V2$(F(2),5)+
      V2$(F(3),1)+V2$(F(3),2)+V2$(F(3),3)         <024>
61185 DR$(II*5-3)=DR$(II*5-3)+V2$(F(3),4)+
      V2$(F(3),5)+V2$(F(4),1)                     <068>
61190 DR$(II*5-2)=V2$(F(4),2)+V2$(F(4),3)+
      V2$(F(4),4)+V2$(F(4),5)+V2$(F(5),1)         <247>
61195 DR$(II*5-2)=DR$(II*5-2)+V2$(F(5),2)+
      V2$(F(5),3)+V2$(F(5),4)                     <240>
61200 DR$(II*5-1)=V2$(F(5),5)+V2$(F(6),1)+
      V2$(F(6),2)+V2$(F(6),3)+V2$(F(6),4)         <031>
61205 DR$(II*5-1)=DR$(II*5-1)+V2$(F(6),5)+
      V2$(F(7),1)+V2$(F(7),2)                     <155>
61210 DR$(II*5-0)=V2$(F(7),3)+V2$(F(7),4)+
      V2$(F(7),5)+V2$(F(8),1)+V2$(F(8),2)         <251>
61215 DR$(II*5-0)=DR$(II*5-0)+V2$(F(8),3)+
      V2$(F(8),4)+V2$(F(8),5)                     <186>
61220 NEXT II                                     <235>
61225 FOR II=1 TO 80                              <120>
61230 PRINT#5,DR$(II):PRINT#4,LE$+CHR$(254
      ):LE$=LE$+" "                               <079>
61235 NEXT II                                     <250>
61240 LE$=""                                      <163>
61245 OPEN 6,4,6:PRINT#6,CHR$(26):PRINT#4,
      :PRINT#6,CHR$(32):CLOSE 6:CLOSE 4:CL
      OSE 5                                       <090>
61250 XX$="":RETURN                               <204>
```


Listing 1. 128/160 Zeichen mit dem Commodore-Drucker MPS 802/1526 (Schluß)

Etiketten wie gedruckt

Mit diesem Programm sind Sie in der Lage, professionell beschriftete Etiketten herzustellen. Und das mit allen ASCII-Zeichen des C 64. Das Programm hat einen eigenen Editor zur Eingabe der Beschriftung. Fertige Labels können gespeichert und geladen werden.

Eine saubere Diskettenbeschriftung macht einiges her. Verschieden gestaltete Labels schaffen Ordnung; nicht nur bei Disketten.

Das Programm ist für einen 1526-Drucker konzipiert, ist aber leicht auf andere Drucker anzupassen. Vorausgesetzt, der Drucker kann die C 64-Grafikzeichen drucken. Wenn nicht, läßt sich das Programm trotzdem einsetzen, wenn man sich auf die Verwendung von Buchstaben beschränkt.

Mit einem Commodore 1526-Drucker können Sie Ihre Aufkleber mit sämtlichen alphanumerischen und C 64-eigenen Zeichen verwenden. Auch reverse und doppelt breite Schrift ist möglich. Das Programm ist für die Verwendung von Endlosetiketten der Größe 88,9x35,5 mm gedacht.

Komfortable Hilfsroutinen

Sehr gut am Programm ist die Tatsache, daß einmal erstellte Aufdrucke gespeichert und wieder geladen werden können. Aber damit sind die Möglichkeiten, die das Programm »Etiketten 64« bietet, noch lange nicht erschöpft. Einige Hilfsfunktionen zur Arbeit mit Disketten runden die »ganze Sache« ab. Man kann sich das Directory zeigen lassen und Befehle an das Diskettenlaufwerk senden. Zum Beispiel "R:" für Rename, "V" für Validate oder "I" zum Initialisieren. Das Programm ist weitestgehend absturzsicher (Ausnahme: Device not Present Error). Fehleingaben können leicht wieder rückgängig gemacht werden: Man gelangt ohne Datenverlust immer wieder ins Hauptmenü zurück.

»Etiketten 64« ist vollständig in Basic geschrieben, um die Anpassung auf andere Drucker zu erleichtern. Deshalb wurde auch der Programmteil DRUCKEN an das Programmende gelegt.

Wenn Sie einen Compiler besitzen, sollten Sie das Basic-Programm damit behandeln. Der Editor wird dann so schnell, daß die Eingabegeschwindigkeit nichts mehr zu wünschen übrigläßt. Das soll aber keinesfalls heißen, das Programm würde ohne Compiler nur bedingt funktionieren. »Etiketten 64« ist auch in nicht compilierter Form ein sehr gutes Programm.

Wenn Sie das Programm starten (sicherheitshalber nach dem Eintippen erst speichern), erscheint das Hauptmenü mit dem Eingabefeld.

Die Menüpunkte:

F1 — Editieren; der Aufdruck kann erstellt werden.

Während des Editierens behalten die Cursortasten ihre volle Funktion bei. Allerdings nur innerhalb des Eingabefeldes. Ebenso funktionieren auch alle anderen Tasten des C 64 zum Editieren. **HOME** setzt den Cursor in die linke obere Ecke, **CLR** löscht das Eingabefeld, **DEL** löscht das Zeichen unterhalb (!) des Cursors. **INST** fügt einen Leerraum an der Cursorposition ein.

Ein Untermenü am unteren Bildschirmrand erleichtert das Editieren noch mit folgenden Funktionen:

Revers schaltet den Reversmodus ein und aus. Der aktuelle Modus wird angezeigt.

Breit schaltet die Breitschrift ein und aus. Doppelt große Zeichen werden durch einen vorangestellten Pfeil gekennzeichnet.

Zeile löschen — eine Zeile wird ab Cursorposition gelöscht. Alle anderen Zeilen rücken nach. Eine komplette Zeile wird gelöscht, wenn der Cursor am Anfang der jeweiligen Zeile steht.

Zeile einfügen — eine Leerzeile wird ab Cursorposition eingefügt. Eine ganze Zeile wird eingefügt, wenn der Cursor, wie bei »Zeile löschen« am Zeilenanfang steht.

Ende — Rücksprung ins Hauptmenü.

F3 — Drucken; das Eingabefeld wird gedruckt. Zum Druck müssen die Etiketten beim 1526-Drucker linksbündig eingelegt werden. Die obere Kante des Aufklebers soll gerade über dem Farbband sichtbar sein.

F5 — speichern; das Eingabefeld wird auf Diskette gespeichert. Wurde der Menüpunkt irrtümlich angewählt, kommt man mit F8 ins Hauptmenü zurück. Vor dem Speichern muß der Bildschirm eingegeben werden. Diskettenfehler werden angezeigt und werden mit RETURN bestätigt. Sollte bereits ein Etikett mit dem eingegebenen Namen auf der Diskette vorhanden sein, so muß man ein Überschreiben mit »J« bestätigen.

F7 — laden; ein Etikettdesign wird von Diskette geladen.

F2 — Directory; die auf einer Diskette gespeicherten Aufkleberbeschriftungen werden aufgelistet.

Weiter holt den nächsten Directory-Eintrag.

Ende stoppt die F2-Funktion. Der zuletzt sichtbare Etikettenname wird in die Ladefunktion übernommen.

F4 — Befehle an das Diskettenlaufwerk.

Hier können Sie Direktbefehle ans Floppylaufwerk schicken. Mit F8 kann diese Funktion abgebrochen werden.

F8 — Ende; beendet einen Programmteil oder das ganze Programm.

Leichte Druckeranpassung

Tabelle 1 zeigt die Variablenliste des Etikettendruckprogramms. Möchten Sie das Listing auf einen anderen Drucker als dem Commodore 1526 anpassen, müssen die Sekundäradressen und Steuercodes entsprechend geändert werden. Sollte eine Anpassung Ihres Druckers bezüglich der Grafikzeichen und doppelt breiter Schrift nicht möglich sein, können Sie das Programm dennoch verwenden. Denn ohne Grafik- und Steuerzeichen sollte das Programm auf jedem Drucker laufen, wenn der Drucker eine Einstellung des Zeilenabstandes zuläßt.

(Gerhard Zinner/hm)

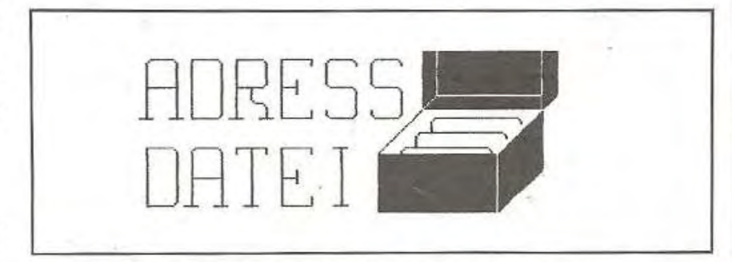


Bild 1. Leicht lassen sich solche Labels drucken

CR$	—	Cursor Right String (zur Cursorpositionierung)
CD$	—	Cursor Down String (zur Cursorpositionierung)
LE$	—	String aus Leerzeichen
T$	—	Textstring (wird aus Eingaben gebildet)
N	—	Pointer f+r Diskettenformatierung
S	—	Cursorspalte
R	—	Cursorreihe
A,L	—	Aktuelle Cursoradresse
RV	—	Pointer für Reversmodus
BR	—	Pointer für Breitmodus
I,J	—	Laufvariablen
UE	—	Pointer für Datei überschreiben
E	—	Fehlernummer der Diskette
E$	—	Fehlertext der Diskette
A$	—	Eingabestring für GET
EN	—	Pointer für Unterprogramm beenden
DE	—	Pointer für Eingabekorrektur
M	—	Max. Eingabelänge
D$	—	Dateiname, Disketten-Kommando
I$	—	Disketten-ID
DA$	—	Dateiname Alt (für Rename)

Tabelle 1. Variablenliste zum Etikettendrucker

```
1000 GOTO 2050                                  <090>
1010 REM                                        <054>
1020 REM                                        <064>
1030 REM            ETIKETTEN 64                <196>
1040 REM            ------------                <125>
1050 REM                                        <096>
1060 REM PROGRAMM ZUM BESCHRIFTEN VON           <078>
1070 REM ENDLOS - ETIKETTEN DER GROESSE         <133>
1080 REM           88.5 * 35.7 MM               <116>
1090 REM                                        <136>
1100 REM                                        <146>
1110 REM                                        <156>
1120 REM        1985  BY   G. ZINNER            <140>
1130 REM                   SCHMALE STR.3        <245>
1140 REM                   4300 ESSEN 11        <251>
1150 REM                                        <196>
1160 REM                                        <206>
1170 REM     ---------------------              <145>
1180 REM     --- ALLG. U'PROGS ---              <007>
1190 REM     ---------------------              <165>
1200 REM                                        <246>
1210 REM                                        <000>
1220 REM         MENUEFELD LOESCHEN             <172>
1230 REM                                        <020>
1240 PRINT LEFT$(CD$,17)                        <076>
1250 FOR I=0 TO 6                               <175>
1260 PRINT LE$                                  <148>
1270 NEXT                                       <008>
1280 RETURN                                     <068>
1290 :                                          <252>
1300 REM         CURSOR SETZEN                  <031>
1310 :                                          <016>
1320 IF S=29 THEN S=28                          <205>
1330 A=1024+((R+3)*40)+S+5                      <213>
1340 IF PEEK(A)>127 THEN POKE A,PEEK(A)-12
     8:RETURN                                   <206>
1350 POKE A,PEEK(A)+128:RETURN                  <188>
1360 :                                          <066>
1370 REM         SCHREIBFELD LOESCHEN           <028>
1380 :                                          <086>
1390 PRINT LEFT$(CD$,4)                         <105>
1400 FOR I=1 TO 11                              <021>
1410 PRINT LEFT$(CR$,6)LEFT$(LE$,28)            <113>
1420 NEXT                                       <160>
1430 RETURN                                     <218>
1440 :                                          <146>
1450 REM         STRING T$ ZUSAMMENBAUEN        <242>
1460 :                                          <166>
1470 RV=0:T$=""                                 <206>
1480 FOR I=L TO L+27                            <015>
1490 A=PEEK(I)                                  <141>
1500 IF RV=0 AND A>128 THEN T$=T$+"{RVSON}
     ":RV=1                                     <116>
1510 IF RV=1 AND A<128 THEN T$=T$+"{RVOFF}
     ":RV=0                                     <202>
1520 IF A>128 THEN A=A-128                      <160>
1530 IF A<32 THEN A=A+64:GOTO 1560              <107>
1540 IF A>63 AND A<96 THEN A=A+32:GOTO 156
     0                                          <246>
1550 IF A>96 THEN A=A+64                        <221>
1560 T$=T$+CHR$(A)                              <014>
1570 NEXT                                       <056>
1580 RETURN                                     <114>
1590 :                                          <042>
1600 REM         DISKETTEN FEHLER               <206>
1610 :                                          <062>
1620 UE=0                                       <016>
1630 INPUT#15,E,E$,T,T                          <111>
1640 IF E=0 THEN RETURN                         <045>
1650 PRINT LEFT$(CD$,25)LEFT$(CR$,2);           <174>
1660 :                                          <112>
1670 IF E<>63 THEN 1730                         <002>
1680 PRINT E;E$;" - AENDERN (J/N)";             <044>
1690 GET A$:IF A$="" THEN 1690                  <092>
1700 IF A$<>"J" THEN 1770                       <055>
1710 UE=1:GOTO 1770                             <216>
1720 :                                          <172>
1730 PRINT E;E$;" - <RETURN>";                  <228>
1740 GET A$                                     <170>
1750 IF A$<>CHR$(13) THEN 1740                  <119>
1760 :                                          <212>
1770 PRINT LEFT$(CD$,25)LEFT$(CR$,2)LEFT$(
     LE$,37);                                   <179>
1780 RETURN                                     <058>
1790 :                                          <242>
1800 REM         DATEINAMEN EINGEBEN            <037>
1810 :                                          <008>
1820 R=19:S=-1:EN=0:DE=0                        <103>
1830 GOSUB 1320                                 <120>
1840 GET A$:IF A$="" THEN 1840                  <243>
1850 GOSUB 1320                                 <140>
1860 IF A$=CHR$(13) THEN 1940                   <254>
1870 IF A$=CHR$(20) THEN DE=1:GOTO 1940         <247>
1880 IF A$=CHR$(140)THEN EN=1:GOTO 1940         <100>
1890 IF A$<CHR$(32) THEN 1830                   <219>
1900 IF A$>CHR$(94) THEN 1830                   <175>
1910 S=S+1:IF S>M THEN S=M                      <240>
1920 PRINT LEFT$(CD$,R+4)LEFT$(CR$,S+4);        <179>
1930 PRINT A$;:GOTO 1830                        <241>
1940 RETURN                                     <220>
1950 :                                          <148>
1960 :                                          <158>
1970 REM     -------------------                <214>
1980 REM     --- ARBEITSFELD ---                <035>
1990 REM     -------------------                <234>
2000 :                                          <198>
2010 :                                          <208>
2020 :                                          <218>
2030 REM         BILDSCHIRMFARBEN               <199>
2040 :                                          <238>
2050 POKE 53280,0:POKE 53281,0:PRINT"{WHIT
     E}"                                        <146>
2060 :                                          <004>
2070 REM         ALLE TASTEN REPEAT             <087>
2080 :                                          <124>
2090 POKE 650,128                               <231>
2100 :                                          <044>
2110 REM         SHIFT+COMMODORE AUS            <027>
2120 :                                          <064>
2130 PRINT CHR$(8)                              <217>
2140 :                                          <084>
2150 REM         FESTE VARIABLEN                <037>
2160 :                                          <104>
2170 CR$="{37RIGHT}"                            <156>
2180 CD$="{HOME,25DOWN}"                        <187>
2190 LE$="{39SPACE}"                            <042>
2200 T$="":N=0                                  <161>
2210 :                                          <154>
2220 REM         BILDSCHIRM AUFBAUEN            <008>
2230 :                                          <174>
2240 PRINT "{CLR,3SPACE}ETIKETTEN 64"SPC(9
     )"(W)1985 BY GZ":PRINT                     <014>
2250 PRINT " U*****************************
     *********I"                                <041>
2260 PRINT " -{3SPACE}t*******************
     **********T{3SPACE}-"                      <227>
```

Listing zu »Etiketten 64«. Beachten Sie bitte den Checksummer 64.

```
2270 FOR I=1 TO 11                               <129>
2280 PRINT " ⊐{2SPACE}-⊐"SPC(28)"⊏-{2SPACE
     }⊐"                                          <228>
2290 NEXT I                                      <086>
2300 PRINT " ⊐{3SPACE}7**********************
     **********Y{3SPACE}⊐"                        <230>
2310 PRINT " J******************************
     *********K"                                  <186>
2320 :                                           <010>
2330 GOSUB 1390                                  <081>
2340 :                                           <030>
2350 REM        MENUE ANZEIGEN                   <096>
2360 :                                           <050>
2370 GOSUB 1240                                  <200>
2380 PRINT LEFT$(CD$,17)                         <202>
2390 PRINT "{3SPACE}@@@@@@@@"                    <200>
2400 PRINT "{3SPACE,RVSON,SPACE}MENUE{SPAC
     E,RVOFF}"                                    <101>
2410 PRINT "{3SPACE}@@@@{16SPACE}@@@@"           <018>
2420 PRINT "{3SPACE,RVSON,SPACE}F1{SPACE,R
     VOFF,SPACE}EDITIEREN{6SPACE,RVSON,SPA
     CE}F2{SPACE,RVOFF,SPACE}DIRECTORY"           <055>
2430 PRINT "{3SPACE,RVSON,SPACE}F3{SPACE,R
     VOFF,SPACE}DRUCKEN{8SPACE,RVSON,SPACE
     }F4{SPACE,RVOFF,SPACE}DISK COM."             <145>
2440 PRINT "{3SPACE,RVSON,SPACE}F5{SPACE,R
     VOFF,SPACE}SPEICHERN{6SPACE}@@@@"            <117>
2450 PRINT "{3SPACE,RVSON,SPACE}F7{SPACE,R
     VOFF,SPACE}LADEN{10SPACE,RVSON,SPACE}
     F8{SPACE,RVOFF,SPACE}ENDE"                   <005>
2460 :                                           <150>
2470 GET A$:IF A$="" THEN 2470                   <013>
2480 IF A$<"{F1}" OR A$>"{F8}" THEN 2470         <207>
2490 :                                           <180>
2500 ON ASC(A$)-132 GOTO 2590,6790,3990,44
     10,4820,5250,2370,6580                       <030>
2510 GOTO 2370                                   <188>
2520 :                                           <210>
2530 :                                           <220>
2540 REM    -----------------                    <157>
2550 REM    --- EDITIEREN ---                    <086>
2560 REM    -----------------                    <179>
2570 :                                           <006>
2580 :                                           <016>
2590 GOSUB 1240                                  <166>
2600 PRINT LEFT$(CD$,17)                         <168>
2610 PRINT "{3SPACE}@@@@@@@@@@@@"                <210>
2620 PRINT "{3SPACE,RVSON,SPACE}EDITIEREN{
     SPACE,RVOFF}"                                <164>
2630 PRINT "{3SPACE}@@@@"SPC(9)"@@@"             <025>
2640 PRINT "{3SPACE,RVSON,SPACE}F1{SPACE,R
     VOFF,SPACE}REVERS >{RVSON}AUS{RVOFF}<
     "                                            <178>
2650 PRINT "{3SPACE,RVSON,SPACE}F3{SPACE,R
     VOFF,SPACE}BREITE >{RVSON}NRM{RVOFF}<
     {3SPACE}@@@@"                                <034>
2660 PRINT "{3SPACE,RVSON,SPACE}F5{SPACE,R
     VOFF,SPACE}ZEILE LOESCH{3SPACE,RVSON,
     SPACE}F6{SPACE,RVOFF,SPACE}ZEILE EINF
     "                                            <055>
2670 PRINT "{23SPACE,RVSON,SPACE}F8{SPACE,
     RVOFF,SPACE}ENDE"                            <119>
2680 BR=0:RV=0                                   <138>
2690 :                                           <126>
2700 REM        CURSOR POSITIONIEREN             <150>
2710 :                                           <146>
2720 R=1:S=1                                     <074>
2730 PRINT LEFT$(CD$,5)LEFT$(CR$,6);             <156>
2740 :                                           <176>
2750 GOSUB 1320                                  <022>
2760 GET A$:IF A$="" OR A$=CHR$(34)THEN 27
     60                                           <245>
2770 GOSUB 1320                                  <042>
2780 :                                           <216>
2790 IF A$>CHR$(31) AND A$<CHR$(128) THEN
     GOSUB 3840:GOTO 2750                         <007>
2800 IF A$>CHR$(160)THEN GOSUB 3840:GOTO 2
     750                                          <168>
2810 IF A$=CHR$(140)THEN 2370                    <220>
2820 IF A$=CHR$(133)THEN GOSUB 3460:GOTO 2
     750                                          <228>
2830 IF A$=CHR$(134)THEN GOSUB 3520:GOTO 2
     750                                          <110>
2840 IF A$=CHR$(135)THEN GOSUB 3580:GOTO 2
     750                                          <130>
2850 IF A$=CHR$(139)THEN GOSUB 3710:GOTO 2
     750                                          <150>
2860 IF A$=CHR$(19) THEN 2720                    <072>
2870 IF A$=CHR$(147)THEN GOSUB 1390:GOTO 2
```

```
     720                                          <105>
2880 IF A$=CHR$(17) THEN GOSUB 3000:GOTO 2
     750                                          <141>
2890 IF A$=CHR$(145)THEN GOSUB 3050:GOTO 2
     750                                          <049>
2900 IF A$=CHR$(29) THEN GOSUB 3100:GOTO 2
     750                                          <230>
2910 IF A$=CHR$(157)THEN GOSUB 3150:GOTO 2
     750                                          <207>
2920 IF A$=CHR$(148)THEN GOSUB 3260:GOTO 2
     750                                          <093>
2930 IF A$=CHR$(20) THEN GOSUB 3360:GOTO 2
     750                                          <117>
2940 IF A$=CHR$(13) THEN GOSUB 3200:GOTO 2
     750                                          <065>
2950 :                                           <132>
2960 GOTO 2750                                   <130>
2970 :                                           <152>
2980 REM        CURSOR DOWN                      <225>
2990 :                                           <172>
3000 R=R+1:IF R>11 THEN R=1                      <050>
3010 RETURN                                      <018>
3020 :                                           <202>
3030 REM        CURSOR UP                        <155>
3040 :                                           <222>
3050 R=R-1:IF R<1 THEN R=11                      <160>
3060 RETURN                                      <068>
3070 :                                           <252>
3080 REM        CURSOR RIGHT                     <245>
3090 :                                           <018>
3100 S=S+1:IF S>28 THEN S=1                      <102>
3110 RETURN                                      <120>
3120 :                                           <048>
3130 REM        CURSOR LEFT                      <058>
3140 :                                           <068>
3150 S=S-1:IF S<1 THEN S=28                      <196>
3160 RETURN                                      <170>
3170 :                                           <098>
3180 REM        RETURN                           <254>
3190 :                                           <118>
3200 R=R+1:IF R>11 THEN R=1                      <252>
3210 S=1                                         <151>
3220 RETURN                                      <230>
3230 :                                           <158>
3240 REM        INSERT                           <100>
3250 :                                           <178>
3260 IF S=28 THEN RETURN                         <124>
3270 A=1024+((R+3)*40)+S+6:E=A+27-S              <116>
3280 FOR I=E TO A STEP-1                         <152>
3290 POKE I,PEEK(I-1)                            <019>
3300 NEXT                                        <006>
3310 POKE I,32                                   <098>
3320 RETURN                                      <074>
3330 :                                           <004>
3340 REM        DELETE                           <035>
3350 :                                           <024>
3360 IF S=28 THEN RETURN                         <096>
3370 A=1024+((R+3)*40)+S+5:E=A+27-S              <154>
3380 FOR I=A TO E                                <188>
3390 POKE I,PEEK(I+1)                            <120>
3400 NEXT                                        <108>
3410 POKE I,32                                   <200>
3420 RETURN                                      <176>
3430 :                                           <104>
3440 REM        REVERS AN / AUS                  <011>
3450 :                                           <124>
3460 PRINT LEFT$(CD$,21)SPC(16)"{RVSON}";        <076>
3470 IF RV=0 THEN PRINT"AN{SPACE,RVOFF}":R
     V=1:RETURN                                   <178>
3480 IF RV=1 THEN PRINT"AUS{RVOFF}":RV=0:R
     ETURN                                        <193>
3490 :                                           <164>
3500 REM        BREITSCHRIFT NRM/DOP             <075>
3510 :                                           <184>
3520 PRINT LEFT$(CD$,22)SPC(16)"{RVSON}";        <140>
3530 IF BR=0 THEN PRINT"DOP{RVOFF}":BR=1:R
     ETURN                                        <143>
3540 IF BR=1 THEN PRINT"NRM{RVOFF}":BR=0:R
     ETURN                                        <224>
3550 :                                           <224>
3560 REM        ZEILE LOESCHEN                   <146>
```

Listing zu »Etiketten 64« (Fortsetzung)

```
3570 :                                     <244>
3580 A=1024+((R+3)*40)+S+5:E=A+28-S        <141>
3590 IF R=11 THEN J=A:GOTO 3660            <173>
3600 A1=(10-R)*40+A                        <057>
3610 FOR J=A TO A1 STEP 40                 <127>
3620 FOR I=J TO E                          <207>
3630 POKE I,PEEK(I+40)                     <213>
3640 NEXT:E=E+40                           <019>
3650 NEXT                                  <104>
3660 FOR I=J TO E:POKE I,32:NEXT           <046>
3670 RETURN                                <172>
3680 :                                     <100>
3690 REM      ZEILE EINFUEGEN              <200>
3700 :                                     <120>
3710 A=1024+((R+3)*40)+S+5+40              <028>
3720 IF R=11 THEN RETURN                   <098>
3730 A1=(10-R)*40+A:E=A1+28-S              <180>
3740 FOR J=A1 TO A STEP -40                <231>
3750 FOR I=J TO E                          <081>
3760 POKE I,PEEK(I-40)                     <088>
3770 NEXT:E=E-40                           <021>
3780 NEXT                                  <234>
3790 FOR I=J TO E:POKE I,32:NEXT           <176>
3800 RETURN                                <046>
3810 :                                     <230>
3820 REM      ZEICHEN SETZEN               <233>
3830 :                                     <250>
3840 IF BR=1 THEN S=S+1                    <101>
3850 S=S+1:IF S>29 THEN S=29               <221>
3860 PRINT LEFT$(CD$,R+4)LEFT$(CR$,S+4);   <087>
3870 IF RV=1 THEN PRINT"{RVSON}";          <191>
3880 IF RV=0 THEN PRINT"{RVOFF}";          <139>
3890 IF BR=1 THEN PRINT"{LEFT}+";          <000>
3900 PRINT A$;                             <115>
3910 RETURN                                <158>
3920 :                                     <086>
3930 :                                     <096>
3940 REM   -----------------               <033>
3950 REM   --- SPEICHERN ---               <026>
3960 REM   -----------------               <053>
3970 :                                     <136>
3980 :                                     <146>
3990 GOSUB 1240                            <040>
4000 PRINT LEFT$(CD$,17)                   <042>
4010 PRINT "{3SPACE}@@@@@@@@@@@@@"SPC(9)"@@@
     @"                                    <156>
4020 PRINT "{3SPACE,RVSON,SPACE}SPEICHERN{
     SPACE,RVOFF}"SPC(9)"{RVSON,SPACE}F8{S
     PACE,RVOFF,SPACE}ENDE":PRINT          <063>
4030 PRINT "{3SPACE}>DATEINAME EINGEBEN"   <107>
4040 PRINT:PRINT "{3SPACE}>{15SPACE}<";    <063>
4050 PRINT "{17LEFT}";                     <072>
4060 :                                     <226>
4070 REM      DATEINAMEN EINGEBEN          <019>
4080 :                                     <246>
4090 IF UE=1 THEN D$="@:"+D$:GOTO 4220     <053>
4100 :                                     <012>
4110 M=14:GOSUB 1820                       <207>
4120 IF EN=1 THEN 2370                     <222>
4130 IF DE=1 THEN 4000                     <048>
4140 :                                     <052>
4150 L=1908:GOSUB 1470                     <177>
4160 D$=LEFT$(T$,15)+"Ŧ"                   <215>
4170 :                                     <082>
4180 REM      BILD ABSPEICHERN             <131>
4190 :                                     <102>
4200 CLOSE 15                              <230>
4210 :                                     <122>
4220 OPEN 15,8,15:OPEN 8,8,8,D$+",S,W"     <016>
4230 GOSUB 1620:IF E<>0 THEN CLOSE 8:CLOSE
      15:GOTO 4090                         <092>
4240 FOR L=1190 TO 1590 STEP 40            <132>
4250 GOSUB 1470                            <174>
4260 :                                     <172>
4270 T$="Ƶ"+T$                             <204>
4280 :                                     <192>
4290 PRINT#8,T$                            <090>
4300 NEXT L                                <088>
4310 CLOSE 8:CLOSE 15                      <214>
4320 :                                     <232>
4330 T$="":GOTO 2370                       <005>
4340 :                                     <252>
4350 :                                     <006>
4360 REM   -------------                   <037>
4370 REM   --- LADEN ---                   <236>
4380 REM   -------------                   <057>
4390 :                                     <048>
4400 :                                     <058>
4410 GOSUB 1240                            <208>
4420 PRINT LEFT$(CD$,17)                   <210>
4430 PRINT "{3SPACE}@@@@@@@@"SPC(13)"@@@@" <112>
4440 PRINT "{3SPACE,RVSON,SPACE}LADEN{SPAC
     E,RVOFF}"SPC(13)"{RVSON,SPACE}F8{SPAC
     E,RVOFF,SPACE}ENDE":PRINT             <028>
4450 PRINT "{3SPACE}>DATEINAME EINGEBEN"   <019>
4460 PRINT:PRINT "{3SPACE}>{15SPACE}<";    <231>
4470 PRINT "{16LEFT}";LEFT$(T$,15)         <033>
4480 :                                     <138>
4490 REM      DATEINAMEN EINGEBEN          <187>
4500 :                                     <158>
4510 M=14:GOSUB 1820                       <097>
4520 IF EN=1 THEN 2370                     <112>
4530 IF DE=1 THEN 4420                     <210>
4540 :                                     <198>
4550 L=1908:GOSUB 1470                     <067>
4560 D$=LEFT$(T$,15)+"Ŧ"                   <105>
4570 :                                     <228>
4580 REM      BILD HOLEN                   <022>
4590 :                                     <248>
4600 CLOSE 15                              <120>
4610 :                                     <014>
4620 GOSUB 1390:R=1                        <237>
4630 OPEN 15,8,15:OPEN 8,8,8,D$+",S,R"     <164>
4640 GOSUB 1620:IF E<>0 THEN CLOSE 8:CLOSE
      15:GOTO 4510                         <164>
4650 INPUT#8,T$                            <118>
4660 :                                     <064>
4670 T$=RIGHT$(T$,LEN(T$)-1)               <163>
4680 :                                     <084>
4690 PRINT LEFT$(CD$,4+R)LEFT$(CR$,6)T$    <117>
4700 R=R+1                                 <036>
4710 IF R<12 THEN 4650                     <255>
4720 CLOSE 8:CLOSE 15                      <116>
4730 :                                     <134>
4740 T$="":GOTO 2370                       <163>
4750 :                                     <154>
4760 :                                     <164>
4770 REM   -----------------               <101>
4780 REM   --- DIRECTORY ---               <223>
4790 REM   -----------------               <121>
4800 :                                     <204>
4810 :                                     <214>
4820 GOSUB 1240                            <108>
4830 PRINT LEFT$(CD$,17)                   <110>
4840 PRINT "{3SPACE}@@@@@@@@@@@"           <152>
4850 PRINT "{3SPACE,RVSON,SPACE}DIRECTORY{
     SPACE,RVOFF}"                         <076>
4860 PRINT "{3SPACE}@@@@"SPC(15)"@@@@"     <146>
4870 PRINT "{3SPACE,RVSON,SPACE}F1{SPACE,R
     VOFF,SPACE}WEITER{8SPACE,RVSON,SPACE}
     F8{SPACE,RVOFF,SPACE}ENDE"            <102>
4880 PRINT:PRINT "{3SPACE}>{15SPACE}<";    <143>
4890 :                                     <040>
4900 REM  REM BILDFILES HOLEN              <084>
4910 :                                     <060>
4920 CLOSE 15                              <188>
4930 :                                     <080>
4940 OPEN 15,8,15                          <016>
4950 OPEN 1,8,0,"$0"                       <018>
4960 :                                     <110>
4970 GOSUB 1620:IF E<>0 THEN CLOSE 1:CLOSE
      15:GOTO 4820                         <191>
4980 :                                     <130>
4990 GET#1,A$,A$,A$,A$,A$,A$               <180>
5000 :                                     <150>
5010 T$=""                                 <033>
5020 GET#1,A$:IF ST<>0 THEN 5150           <215>
5030 IF A$<>CHR$(34) THEN 5020             <039>
5040 GET#1,A$:IF A$<>CHR$(34) THEN T$=T$+A
     $:GOTO 5040                           <227>
5050 IF RIGHT$(T$,1)<>CHR$(166) THEN 5120  <129>
5060 :                                     <210>
5070 PRINT LEFT$(CD$,23)LEFT$(CR$,4)LEFT$(
     T$,15)                                <212>
5080 :                                     <230>
5090 GET A$:IF A$="" THEN 5090             <189>
5100 IF A$=CHR$(140) THEN 5150             <191>
5110 :                                     <004>
5120 GET#1,A$:IF A$<>"" THEN 5120          <171>
5130 IF ST=0 THEN 5010                     <078>
```

Listing zu »Etiketten 64« (Fortsetzung)

```
5140 :                                           <036>
5150 CLOSE 1:CLOSE 15                            <238>
5160 :                                           <056>
5170 GOTO 2370                                   <054>
5180 :                                           <076>
5190 :                                           <086>
5200 REM    -------------------                  <142>
5210 REM    --- DISKCOMANDS ---                  <064>
5220 REM    -------------------                  <162>
5230 :                                           <126>
5240 :                                           <136>
5250 GOSUB 1240                                  <030>
5260 PRINT LEFT$(CD$,17)                         <032>
5270 PRINT "{3SPACE}▀▀▀▀▀▀▀▀▀▀▀▀▀▀"SPC(7)"
     ▀▀▀▀"                                       <055>
5280 PRINT "{3SPACE,RVSON,SPACE}DISK-COMAN
     DS{SPACE,RVOFF}"SPC(7)"{RVSON,SPACE}F
     8{SPACE,RVOFF,SPACE}ENDE"                   <022>
5290 PRINT "{3SPACE}▀▀▀"SPC(18)"▀▀▀"             <229>
5300 PRINT "{3SPACE,RVSON,SPACE}N{SPACE,RV
     OFF,SPACE}NEW (FORMAT.){4SPACE,RVSON,
     SPACE}S{SPACE,RVOFF,SPACE}SCRATCH"          <229>
5310 PRINT "{3SPACE,RVSON,SPACE}V{SPACE,RV
     OFF,SPACE}VALIDATE{9SPACE,RVSON,SPACE
     }R{SPACE,RVOFF,SPACE}RENAME"                <060>
5320 PRINT "{3SPACE,RVSON,SPACE}I{SPACE,RV
     OFF,SPACE}INITIALIZE"                       <025>
5330 :                                           <226>
5340 CLOSE 15                                    <098>
5350 :                                           <246>
5360 GET A$:IF A$="" THEN 5360                   <012>
5370 :                                           <010>
5380 IF A$=CHR$(140) THEN 2370                   <252>
5390 IF A$="N" THEN 5480                         <038>
5400 IF A$="S" THEN 5850                         <162>
5410 IF A$="V" OR A$="I" THEN 6040               <193>
5420 IF A$="R" THEN 6140                         <242>
5430 :                                           <072>
5440 GOTO 5360                                   <062>
5450 :                                           <092>
5460 REM       NEW (FORMATIEREN)                 <056>
5470 :                                           <112>
5480 GOSUB 1240                                  <006>
5490 PRINT LEFT$(CD$,17)                         <008>
5500 PRINT "{3SPACE}▀▀▀"                         <053>
5510 PRINT "{3SPACE,RVSON,SPACE}N{SPACE,RV
     OFF,SPACE}NEW (FORMATIEREN)":PRINT          <068>
5520 PRINT "{3SPACE}>DISKETTENNAME":PRINT        <120>
5530 PRINT "{3SPACE}>{16SPACE}<";                <202>
5540 PRINT "{16LEFT}";                           <196>
5550 :                                           <192>
5560 M=15:GOSUB 1820                             <163>
5570 IF EN=1 THEN 2370                           <146>
5580 IF DE=1 THEN 5490                           <017>
5590 :                                           <232>
5600 L=1908:GOSUB 1470                           <101>
5610 IF LEFT$(T$,1)=" " THEN 2370                <227>
5620 D$=LEFT$(T$,16)                             <075>
5630 :                                           <016>
5640 PRINT LEFT$(CD$,21)LEFT$(CR$,25)", ID
     "                                           <014>
5650 PRINT:PRINT SPC(26) ">{2SPACE}<"            <241>
5660 R=19:S=22:EN=0:DE=0:M=24                    <035>
5670 GOSUB 1830                                  <006>
5680 IF EN=1 THEN 2370                           <002>
5690 IF DE=1 THEN 5640                           <113>
5700 :                                           <088>
5710 L=1931:GOSUB 1470                           <017>
5720 IF LEFT$(T$,1)=" " THEN 2370                <083>
5730 I$=LEFT$(T$,2)                              <217>
5740 :                                           <128>
5750 PRINT LEFT$(CD$,20) "{3SPACE}▀▀▀▀▀▀▀▀
     ▀▀▀▀▀▀▀▀▀▀▀▀▀▀▀▀▀▀▀▀▀▀"                      <226>
5760 PRINT "{3SPACE,RVSON,SPACE}NEUE DISKE
     TTE EINGELEGT (J/N){SPACE,RVSON}"           <119>
5770 :                                           <158>
5780 GET A$:IF A$="" THEN 5780                   <182>
5790 IF A$<>"J" THEN 2370                        <025>
5800 :                                           <188>
5810 D$="N0:"+D$+","+I$:GOTO 6450                <118>
5820 :                                           <208>
5830 REM       SCRATCHEN                         <028>
5840 :                                           <228>
5850 GOSUB 1240                                  <122>
5860 PRINT LEFT$(CD$,17)                         <124>
5870 PRINT "{3SPACE}▀▀▀"                         <169>
5880 PRINT "{3SPACE,RVSON,SPACE}S{SPACE,RV
     OFF,SPACE}SCRATCH":PRINT                    <124>
5890 PRINT "{3SPACE}>DATEINAME EINGEBEN":P
     RINT                                        <117>
5900 PRINT "{3SPACE}>{15SPACE}<";                <064>
5910 PRINT "{16LEFT}"LEFT$(T$,15);               <002>
5920 :                                           <139>
5930 M=14:GOSUB 1820                             <249>
5940 IF EN=1 THEN 2370                           <008>
5950 IF DE=1 THEN 5860                           <131>
5960 :                                           <094>
5970 L=1908:GOSUB 1470                           <219>
5980 IF LEFT$(T$,1)=" " THEN 2370                <089>
5990 D$="S0:"+LEFT$(T$,15)+"╤"                   <204>
6000 GOTO 6450                                   <106>
6010 :                                           <144>
6020 REM       VALIDATE/INITIALIZE               <150>
6030 :                                           <164>
6040 GOSUB 1240                                  <058>
6050 PRINT LEFT$(CD$,17)                         <060>
6060 PRINT "{3SPACE}▀▀▀"                         <105>
6070 PRINT "{3SPACE,RVSON,SPACE}"A$"{SPACE
     ,RVOFF,SPACE}VALIDATE / INITIALIZE":P
     RINT                                        <214>
6080 :                                           <214>
6090 D$=A$+"0"                                   <229>
6100 GOTO 6450                                   <206>
6110 :                                           <244>
6120 REM       RENAME                            <062>
6130 :                                           <008>
6140 GOSUB 1240                                  <158>
6150 PRINT LEFT$(CD$,17)                         <162>
6160 PRINT "{3SPACE}▀▀▀"                         <207>
6170 PRINT "{3SPACE,RVSON,SPACE}R{SPACE,RV
     OFF,SPACE}RENAME":PRINT                     <059>
6180 PRINT "{3SPACE}>ALTER DATEINAME":PRIN
     T                                           <084>
6190 PRINT "{3SPACE}>{15SPACE}<";                <100>
6200 PRINT "{16LEFT}"LEFT$(T$,15);               <038>
6210 :                                           <090>
6220 M=14:GOSUB 1820                             <029>
6230 IF EN=1 THEN 2370                           <044>
6240 IF DE=1 THEN 6150                           <150>
6250 :                                           <130>
6260 L=1908:GOSUB 1470                           <255>
6270 DA$=LEFT$(T$,15)+"╤"                        <153>
6280 :                                           <160>
6290 PRINT LEFT$(CD$,19)                         <054>
6300 PRINT "{4SPACE}▀▀▀▀▀"                       <132>
6310 PRINT "{3SPACE}>{RVSON}NEUER{RVOFF,SP
     ACE}DATEINAME":PRINT                        <204>
6320 PRINT "{3SPACE}>{15SPACE}<";                <230>
6330 PRINT "{16LEFT}";                           <224>
6340 M=14:GOSUB 1820                             <149>
6350 IF EN=1 THEN 2370                           <164>
6360 IF DE=1 THEN 6290                           <032>
6370 :                                           <250>
6380 L=1908:GOSUB 1470                           <119>
6390 D$=LEFT$(T$,15)+"╤"                         <157>
6400 :                                           <026>
6410 D$="R0:"+D$+"="+DA$                         <100>
6420 :                                           <046>
6430 REM       DISKBEFEHL AUSFUEHREN             <090>
6440 :                                           <066>
6450 OPEN 15,8,15                                <002>
6460 GOSUB 1620:IF E<>0 THEN CLOSE 15:GOTO
      2370                                       <171>
6470 PRINT#15,D$                                 <094>
6480 GOSUB 1620:IF E<>0 THEN CLOSE 15            <111>
6490 :                                           <116>
6500 T$="":GOTO 2370                             <145>
6510 :                                           <136>
6520 :                                           <146>
6530 REM    --------------------                 <021>
6540 REM    --- PROG BEENDEN ---                 <190>
6550 REM    --------------------                 <041>
6560 :                                           <186>
6570 :                                           <196>
6580 GOSUB 1240                                  <090>
6590 PRINT LEFT$(CD$,17)                         <092>
6600 PRINT "{3SPACE}▀▀▀▀▀▀"                      <063>
6610 PRINT "{3SPACE,RVSON,SPACE}ENDE{SPACE
     ,RVOFF}":PRINT                              <085>
6620 PRINT "{3SPACE}SIND SIE SICHER (J/N)
     :"                                          <200>
```

Listing zu »Etiketten 64«. Beachten Sie bitte den Checksummer 64. (Fortsetzung)

```
6630 GET A$:IF A$="" THEN 6630                  <107>
6640 IF A$<>"J" THEN 2370                       <113>
6650 PRINT "{CLR}OK - BIS BALD!":END            <054>
6660 :                                          <032>
6670 :                                          <042>
6680 REM    ----------------------              <112>
6690 REM    ---       DRUCKEN     ---           <074>
6700 REM    ----------------------              <132>
6710 REM    - 1526 MIT NEUEM ROM -              <150>
6720 REM    -         IM 1525 MODUS -           <225>
6730 REM    -                    -              <247>
6740 REM    - CHR$(14)=BREIT EIN -              <223>
6750 REM    - CHR$(15)=BREIT AUS -              <142>
6760 REM    ----------------------              <192>
6770 :                                          <142>
6780 :                                          <152>
6790 GOSUB 1240                                 <046>
6800 PRINT LEFT$(CD$,17)                        <048>
6810 PRINT "{3SPACE}@@@@@@@@@@"                 <208>
6820 PRINT "{3SPACE,RVSON,SPACE}DRUCKEN{SP
     ACE,RVOFF}"                                <003>
6830 PRINT:PRINT "{3SPACE}DRUCKER EINGESCH
     ALTET (J/N) : "                            <236>
6840 GET A$:IF A$="" THEN 6840                  <065>
6850 IF A$<>"J" THEN GOTO 2370                  <073>
6860 :                                          <232>
6870 REM        DRUCKKANAL OEFFNEN              <072>
6880 :                                          <252>
6890 OPEN 3,4                                   <125>
6900 :                                          <016>
6910 REM        ZEILENABSTAND = 0               <166>
6920 :                                          <038>
6930 OPEN 6,4,6:PRINT#6,CHR$(20):CLOSE 6        <126>
6940 :                                          <058>
6950 REM        ZEILEN HOLEN                    <056>
6960 :                                          <078>
6970 FOR L=1190 TO 1590 STEP 40                 <068>
6980 GOSUB 1470                                 <110>
6990 :                                          <108>
7000 REM        DRUCKSTRING ERSTELLEN           <119>
7010 :                                          <128>
7020 FOR J=1 TO LEN(T$)                         <254>
7030 A$=MID$(T$,J,1)                            <200>
7040 IF A$<>"←" THEN PRINT#3,A$;:GOTO 7080      <230>
7050 PRINT#3,CHR$(14);                          <137>
7060 J=J+1:A$=MID$(T$,J,1)                      <109>
7070 PRINT#3,A$;CHR$(15);                       <220>
7080 NEXT J                                     <058>
7090 :                                          <208>
7100 REM        STRING DRUCKEN                  <169>
7110 :                                          <228>
7120 PRINT#3                                    <001>
7130 NEXT L                                     <124>
7140 :                                          <002>
7150 REM        ZEILENVORSCHUB                  <202>
7160 :                                          <022>
7170 FOR I=1 TO 5:PRINT#3:NEXT I                <247>
7180 :                                          <044>
7190 :      REM DRUCKKANAL SCHLIESSEN           <230>
7200 :                                          <144>
7210 CLOSE 3                                    <125>
7220 :                                          <084>
7230 T$="":GOTO 2370                            <113>
```

© 64'er

Listing zu »Etiketten 64«. Beachten Sie bitte den Checksummer 64. (Schluß)

Die entscheidende Verbesserung

Erweitern Sie die Eyssele-Schnittstelle um eine Hardcopy-Funktion. Sie können dann mit einem einfachen Befehl genau das drucken, was am Bildschirm zu sehen ist.

Diese Erweiterung verbessert die Eyssele-Schnittstelle (auf Seite 22) gewaltig. Denn es wird nicht nur eine Text-Hardcopy implementiert, sondern die Eyssele-Schnittstelle kann damit auch einfach auf andere Drucker als den Epson RX/FX-Typen angepaßt werden. Die abgedruckte Version läuft mit einem CP-80 über den Centronics-Eingang. Zur Anpassung auf die Epson-Drucker sind folgende Änderungen nötig:

60 DZ=5

61 D(1)=27:D(2)=42:D(3)=4:D(4)=8:D(5)=0

Zum Anschluß an den C 64 müssen Sie sich nur ein Verbindungskabel vom User-Port zum Centronics-Eingang des Druckers herstellen (lassen). Welche Pins Sie mit welchen verbinden müssen, lesen Sie bitte im vorhergehenden Artikel.

So ist die Erweiterung zu verwenden

Tippen Sie die Eyssele-Schnittstelle (»Die billigste Centronics-Schnittstelle ...«) mit dem MSE ab und speichern Sie das Maschinenprogramm auf Kassette oder Diskette. Dann laden Sie es mit LOAD "CENTRONICS",8,1 und geben NEW ein, damit die Basis-Vektoren wieder ordentlich gesetzt werden. Dann können Sie diese Erweiterung mit dem Checksummer aus dieser Ausgabe in den C 64 eingeben. Bevor Sie allerdings RUN eintippen, sollten Sie das Basic-Programm sicherheitshalber speichern.

Während des Programmlaufs werden Sie gefragt, ob eine lauffähige Version der erweiterten Schnittstelle als Maschinenprogramm auf Diskette oder Kassette gespeichert werden soll. Reagieren Sie bitte mit D (Diskette), K (Kassette) oder N (nicht speichern). Das erzeugte Programm bekommt den Namen »Druckertreiber«. Es muß mit LOAD "DRUCKERTREIBER",8,1 oder 1,1 geladen werden. Danach ist wieder das gewohnte NEW erforderlich, die Schnittstelle wird mit SYS 52200 aktiviert.

Bevor Sie den Computer und Drucker einschalten, sollten Sie erst das Kabel anschließen. Denn beim Aufstecken des User-Port-Steckers kann man leicht einen Kurzschluß erzeugen. Im Glücksfall brennt dann die Sicherung im C 64 durch. Meist verabschiedet sich aber eine der beiden CIA-Bausteine und eine nicht billige Reparatur ist die Folge.

Es gelten die Geräte- und Sekundäradressen der Eyssele-Schnittstelle. Lesen Sie diese bitte dort in Tabelle 1 nach.

Originalgetreue Kopie des Bildschirmes

Nun zur Praxis. Durch Aufruf des Hardcopy-Programmteils wird eine absolut originalgetreue Kopie des normalen Text-Bildschirms gedruckt. Auch reverse Zeichen werden berücksichtigt. Sie können den Abstand zwischen den Druckzeilen wählen. Ohne Abstand, wie am Bildschirm (ideal für Blockgrafiken), oder mit Abstand wie in einem Listing. Die Befehle für die Hardcopy:

SYS 52270,page:druckt mit

SYS 52408,page:ohne Abstand

Bei Page müssen Sie angeben, welche Speicherseite gedruckt werden soll. Der Bildschirmspeicher muß sich nämlich nicht immer zwischen den Adressen 1024 und 2023 (Page=4) befinden. Durch Aufruf der Hardcopy wird die Schnittstelle gleichzeitig auch initialisiert. Sie können sich dann also SYS 52200 (auch SYS 51456 ist möglich) sparen.

Reverse Zeichen sind kein Problem

Es werden auch reverse Grafikzeichen im Groß-/Kleinschriftmodus gedruckt. Das sind alle Grafikzeichen, die über die CBM-Taste erreichbar sind.

Zeichen mit einem Code von 91 bis 95 werden im Grafikmodus gedruckt, wie andere Sonderzeichen auch. Der Grund dafür ist, daß die C 64-Zeichen in diesem Bereich meist nicht mit den Druckerzeichen in diesem Bereich übereinstimmen. (Dipl.-Ing. Wilhelm Meierhofer/hm)

Eine Hardcopy des Textbildschirms mit Grafikzeichen. Es muß nicht immer HiRes sein. Sie sehen den Bildschirmaufbau des Etikettendruck-Programmes.

```
10 REM ERWEITERUNG ZUM TREIBERPROGRAMM FUE
   R CENTRONICS-DRUCKER VON H EYSSELE AUS   <054>
12 REM 64-ER NR. 7/84                        <198>
14 REM                                       <076>
16 REM VON W. MEIERHOFER                     <220>
18 REM                                       <080>
20 REM ZUSAETZLICHE FAEHIGKEITEN:            <249>
22 REM    - HARCOPY MIT ZEILENABSTAND        <098>
24 REM    - HARCOPY BILDSCHIRM ABSOLUT       <122>
26 REM    - ERKENNUNG UND AUSDRUCK VON REVE
   RSEN SONDERZEICHEN                        <134>
30 REM NAEHERES ENTNEHMEN SIE BITTE DER BE
   IGEFUEGTEN ANLEITUNG                      <142>
32 REM                                       <031>
34 REM                                       <096>
40 PRINT"{CLR}DAS MASCHINENPROGRAMM VON H.
    EYSSELE"                                 <204>
42 PRINT"MUSS SICH BEREITS IM SPEICHER BEF
   INDEN !"                                  <249>
44 PRINT"HABEN SIE DIE DRUCKERSTEUERCODES
   IN"                                       <056>
46 PRINT"ZEILE 60 -65 AN IHREN DRUCKER ANG
   EPASST?"                                  <247>
50 PRINT"WEITER MIT IRGENDEINER TASTE !"    <007>
52 IF PEEK(203)=64 THEN 52                   <152>
54 PRINT:PRINT"BITTE WARTEN":PRINT:PRINT     <122>
55 REM                                       <117>
56 REM                                       <118>
58 REM ***FOLGENDE PARAMETER SIND AUF DEN
   DRUCKER CP80 ABGESTIMMT***                <186>
59 REM BITTE AENDERN SIE SIE FUER IHREN DR
   UCKER                                     <034>
60 DZ=4:REM ANZAHL NOTWENDIGER CODES (MAX
   9), UM 8 BYTES GRAFIK AUSZUGEBEN          <121>
61 D(1)=27:D(2)=75:D(3)=8:D(4)=0:REM BITTE
   AENDERN ODER ERGAENZEN                    <185>
62 NZ=3:REM ANZAHL NOTWENDIGER CODES (MAX
   8), FUER NULL-ZEILENABSTAND               <101>
63 N(1)=27:N(2)=65:N(3)=7:REM BITTE AENDER
   N ODER ERGAENZEN                          <216>
64 GZ=2:REM ANZAHL NOTWENDIGER CODES (MAX
   5), FUER GRUNDZEILENABSTAND               <080>
65 G(1)=27:G(2)=50:REM BITTE AENDERN ODER
   ERGAENZEN                                 <195>
66 REM                                       <128>
67 REM                                       <129>
68 REM ***EINPOKEN VON DATAS***              <081>
70 FOR I=52200 TO 52267:READ X:POKE I,X:FE
   =FE+X:NEXT                                <077>
71 IF FE<>8381 THEN PRINT"FEHLER IM ERSTEN
    DATENBLOCK":END                          <207>
72 FOR I=52270 TO 52426:READ X:POKE I,X:FA
   =FA+X:NEXT                                <133>
73 IF FA<>22642 THEN PRINT"FEHLER IM ZWEIT
   EN DATENBLOCK":END                        <102>
75 REM                                       <137>
76 REM                                       <138>
77 REM ***DRUCKERSPEZIFISCHE PARAMETERN***   <245>
80 POKE 51992,DZ:FOR I=1 TO DZ:POKE 52177+
   I,D(I):NEXT                               <076>
82 POKE 52421,NZ:FOR I=1 TO NZ:POKE 52191+
   I,N(I):NEXT                               <217>
84 POKE 52404,GZ:FOR I=1 TO GZ:POKE 52186+
   I,G(I):NEXT                               <233>
86 REM                                       <148>
87 REM                                       <149>
88 REM ***ANSCHLIESSEN DER ERWEITERUNG***    <255>
90 POKE 51758,21:POKE 51760,91:POKE 51770,
   9:POKE 51772,219:POKE 51774,5             <054>
91 POKE 51780,76:POKE 51781,235:POKE 51782
   ,203:POKE 51783,234:POKE 51785,91         <177>
92 POKE 51922,76:POKE 51923,25:POKE 51924,
   204                                       <184>
96 REM                                       <158>
97 REM                                       <159>
98 REM------------------------------------
   ------------------------------------      <183>
99 REM 1.DATENBLOCK                          <136>
100 DATA 76,0,201,164,199,208,19,201,18,24
    0,10,201,255,240,3,76,72,202,76          <195>
101 DATA 96,202,160,1,132,199,96,201,146,2
    40,7,201,13,240,8,76,79,202,160          <230>
102 DATA 0,132,199,96,160,0,132,199,76,72,
    202,74,74,74,72,165,199,208,4,104        <088>
103 DATA 76,213,202,104,24,105,4,76,213,20
    2                                        <157>
199 REM 2.DATENBLOCK                         <244>
200 DATA 32,0,201,32,253,174,32,158,183,16
    9,255,133,249,202,138,133,250,232        <041>
201 DATA 232,232,232,138,133,248,162,0,134
    ,247,160,0,230,249,208,2,230,250         <205>
202 DATA 230,247,177,249,201,128,144,21,17
    2,24,208,192,21,208,4,160,208,208        <216>
203 DATA 2,160,216,132,6,32,196,202,76,143
    ,204,201,32,176,5,9,64,76,127,204        <236>
204 DATA 201,64,144,3,24,105,128,172,24,20
    8,192,21,208,6,32,68,202,76,143          <191>
205 DATA 204,32,43,202,166,247,224,40,208,
    181,162,0,134,247,169,13,32,64,201       <121>
206 DATA 165,250,197,248,208,166,165,249,2
    01,231,208,160,162,0,189,219,203         <249>
207 DATA 32,64,201,232,224,2,208,245,96,32
    ,0,201,162,0,189,224,203,32,64,201       <047>
208 DATA 232,224,3,208,245,76,49,204         <227>
210 REM                                      <016>
220 REM                                      <026>
230 REM                                      <036>
300 PRINT"PROGRAMM ALS MASCHINEN-PROGRAMM
    AUF{5SPACE}DISKETTE/CASSETTE LEGEN D/C
    /N?"                                     <179>
310 GET A$:IF A$=""THEN 310                  <057>
320 IF A$="D"THEN SYS(57812)"DRUCKTREIBER"
    ,8:GOTO 360                              <214>
330 IF A$="C"THEN SYS(57812)"DRUCKTREIBER"
    ,1:GOTO 360                              <081>
340 IF A$="N"THEN 400                        <154>
350 GOTO 310                                 <056>
360 POKE 193,0:POKE 194,201:POKE 174,203:P
    OKE 175,204:SYS 62957                    <187>
370 PRINT:PRINT"PROGRAMM KANN JETZT MIT"     <229>
380 PRINT"LOAD'DRUCKTREIBER',8,1 (BZW. {2SP
    ACE},1,1)"                               <243>
390 PRINT"DIREKT GELADEN WERDEN !{2SPACE}(
    ABER DABEI"                              <135>
392 PRINT"'NEW' + INITIALISIERUNG NICHT VE
    RGESSEN)"                                <007>
400 SYS 52200:END                            <125>
```


Listing zur Eyssele-Erweiterung. Beachten Sie bitte die Eingabehinweise. Zu diesem Programm benötigen Sie die schon beschriebene Eyssele-Schnittstelle.

Die billigste Centronics-Schnittstelle für den C 64

Diese Schnittstelle übertrifft mit ihren Leistungsmerkmalen viele käufliche Hardware-Schnittstellen um ein Vielfaches. Nicht aber im Preis: Nur etwa 15 Mark kostet diese »Selbstbau«-Centronics-Schnittstelle.

Leser des 64'er Magazins, die schon »seit Anfang dabei sind«, erkennen diese Schnittstelle bestimmt wieder. In Ausgabe 7/85 wurde sie unter dem Namen Eyssele-Schnittstelle vorgestellt.

Wie im Fernsehen! Jetzt fangen die auch schon mit Wiederholungen an! Diese Vermutung trifft nur in diesem speziellen Fall zu. Wir haben guten Grund, das Programm zu der Eyssele-Schnittstelle nochmals abzudrucken. Denn viele unserer Lesereinsendungen — auch einige Listings in diesem Sonderheft — beziehen sich auf diese Software-Schnittstelle. Die alten Hasen unter Ihnen mögen uns verzeihen. Aber seit einem Jahr wuchs unsere Leserzahl so gewaltig, daß ein großer Teil die Eyssele-Schnittstelle einfach noch nicht kennt.

Es handelt sich hierbei um ein Maschinensprache-programm mit 726 Byte Länge, das den Adreßbereich C900-CBD1 (dezimal: 51456 bis 52182) belegt (Listing 1). Damit kann zum Beispiel das DOS 5.1 auf der Demodiskette ohne Probleme zusätzlich geladen werden.

Das Programm ist ohne Einschränkung für alle Epson-kompatiblen Drucker anwendbar, die über einen Bitmustermodus verfügen. Dieser Modus wird für die Ausgabe der Commodore-eigenen Grafikzeichen benötigt. Das Programm besteht aus mehreren Programmteilen, von denen die meisten Erweiterungen bestehender Ein-/Ausgaberoutinen des Betriebssystems sind und bei der Initialisierung in diese eingebunden werden. Dadurch können schon bestehende Basic-Befehle wie OPEN und PRINT # zum Drucken benutzt, und Programme müssen nicht umgeschrieben werden. Zur Ansteuerung verschiedener Druckmodi wurden jedoch zusätzliche Gerätenummern definiert, deren Bedeutungen in Tabelle 1 erklärt sind. Gerätenummer 16 realisiert einen sogenannten Direktmodus, mit dem die internen Commodore-Zeichencodes ohne Wandlung an den Drucker gelangen. Der Direktmodus ist zur Ausgabe von Steuerzeichen oder bei der Verwendung des Druckers als Plotter zur Einzelnadelsteuerung beziehungsweise zur Ausgabe von Bitmustern geeignet. Der Textmodus (Groß- und Kleinschreibung) ist unter Gerätenummer 18 und, weil er wohl am häufigsten bei bereits bestehenden Programmen benutzt wird, unter Gerätenummer 4 ansprechbar. Die Gerätenummer 19 realisiert den Großschrift/Grafik-Modus, wie er beim C 64 gleich nach dem Einschalten voreingestellt ist. Der wichtige Modus zum Listen von Programmen wurde mit Gerätenummer 17 realisiert. Es läßt sich über die Sekundäradresse, wie in den Musterlistings (Bild 1 bis 3) gezeigt, noch zwischen vier Fällen unterscheiden. Einmal kann ausgewählt werden, ob das Listing, wie vom Bildschirm her gewohnt, mit Großbuchstaben und Grafikzeichen oder im Textmodus mit großen und kleinen Buchstaben gedruckt wird. Zum anderen kann man auswählen, ob die Steuerzeichen wie bei der Bildschirmausgabe als inverse Zeichen oder durch Abkürzungen wie <CRD> (Cursor down) im Klartext gedruckt werden. Durch all diese Möglichkeiten kann der Anwender zum einen auf seinem Drucker Ausgaben erzeugen, wie man sie von Commodore-Druckern her gewohnt ist, zum anderen auch alle Möglichkeiten seines Druckers voll nutzen.

Commodore-eigene Grafikzeichen

Die Ausgabe von Grafikzeichen erfolgt in der Routine OUTGEN ab Adresse $CAC4. Zu diesem Zweck wird der Zeichengenerator des C 64 ausgelesen. Die Zeichen, die aus einer 8x8-Punktmatrix bestehen, sind dort in je 8 Byte abgelegt. Jedes Byte repräsentiert das Punktmuster einer Zeile der Matrix. Ein Matrixdrucker druckt die Zeichen, indem er sie ebenso als Muster von matrixartig angeordneten Punkten zu Papier bringt. Jedoch gibt er die Punktmatrix nicht wie der Bildschirm zeilenweise, sondern spaltenweise aus. Deshalb wird in OUTGEN die im Zeichengenerator in 8 Zeilenbyte gespeicherte Zeichenmatrix zunächst in 8 Spaltenbyte umorganisiert. Nach Umschalten des Druckers in den sogenannten Bitmustermodus, in dem er jedes ankommende Datenbyte nicht mehr als ASCII-Zeichen, sondern als Musterbyte für eine Matrixspalte interpretiert, gibt OUTGEN die acht geänderten Zeichenbyte an den Drucker aus. Dieser fügt sie wieder zu einem 8x8-Punktmatrixzeichen zusammen. So ist es möglich, auf einem Matrixdrucker exakte Kopien der C 64-Bildschirmzeichen herzustellen, obwohl dieser nicht über den entsprechenden Zeichensatz verfügt.

So benutzen Sie das Programm

Das Listing sollten Sie mit dem MSE eingeben und auf Diskette oder Kassette speichern, bevor Sie es starten. Bevor Sie es allerdings laden, sollten Sie sich das Verbindungskabel User-Port-Centronics-Buchse zusammenlöten. Dazu ist in Tabelle 2 ein Verbindungsplan angegeben. Das Kabel sollte für eine störungsfreie Funktion nicht länger als ein Meter lang sein und aus einem abgeschirmten, mehradrigen Steuerkabel bestehen, das man in (fast) jedem Elektronik-

GERÄTENUMMER 16		=	DIREKTMODUS
GERÄTENUMMER 18,4		=	TEXTMODUS
GERÄTENUMMER 19		=	GRAFIKMODUS
GERÄTENUMMER 17		=	LIST-MODUS
SEK-ADR.	MODUS		STEUERZEICHEN
0	NORMAL		NORMAL
1	KLEIN		NORMAL
2	NORMAL		ERKLÄRT
3	KLEIN		ERKLÄRT

Tabelle 1. Die Bedeutung der Gerätenummern

USER PORT	—	CENTRONICS
A	GND	16
B	FLAG — BUSY	11
C	DO	2
D	D1	3
E	D2	4
F	D3	5
H	D4	6
J	D5	7
K	D6	8
L	D7	9
M	PA2 — STROBE	1

Tabelle 2. Der Verkabelungsplan

Bastelgeschäft findet. Dort sind auch meist der Centronics-Stecker und der Stecker für den User-Port erhältlich.

Initialisiert wird die geladene Treiberroutine mit SYS 12*4096+9*256 beziehungsweise SYS 51456.

Hierbei wird die Routine in das Betriebssystem eingebunden. Jedoch Vorsicht: Nach einem Break, zum Beispiel durch die Betätigung der Tasten RUN/STOP und RESTORE ausgelöst, muß die Routine erneut initialisiert werden, da die I/O-Vektoren vom Betriebssystem zurückgesetzt wurden.

Die einzelnen Druckmodi spricht man mit den üblichen Basicbefehlen an. Geöffnet wird der Ausgabekanal mit:

OPEN log. Dateinummer, Geräteadr. [,Sekundäradr.]

Die eckigen Klammern kennzeichnen optionale Angaben. So kann dann auf den geöffneten Kanal mit PRINT# log. Dateinummer ausgegeben werden. Ein Programmlisting (Bild 2) wird zum Beispiel erzeugt mit den Befehlen

OPEN 17,17 [,Sekundäradr.]:CMD17:
LIST
PRINT#17:CLOSE17

Der PRINT-Befehl vor dem CLOSE ist notwendig, damit der CMD-Modus aufgehoben wird.

Umstellen auf beliebige Drucker mit Centronics-Schnittstelle

Das Programm wurde für einen Epson-Drucker geschrieben. Unverändert ist es für jeden anderen Drucker verwendbar, sofern auf die Ausgabe von Commodore-eigenen Grafikzeichen verzichtet wird. Für diese Funktion muß der Drucker vorübergehend in den Bitmustermodus zurückgeschaltet werden, was während einer normalen Textausgabe möglich sein muß. Diese Umschaltung erfolgt im Programm in der Schleife ab Adresse $CB10. Die Anzahl der auszugebenden Steuerzeichen ist unter Adresse $CB18 gespeichert und die Steuerzeichen selbst stehen ab Adresse $CBD2 am Ende des Programms. Für einen Epson-Drucker wird die Folge

ESC*480

ausgegeben, wobei mit 4 der Bitmustermodus »CRT-Grafik« ausgewählt wird, 8 das niederwertige Byte und 0 das höherwertige Byte der Anzahl auszugebender Punktmatrixspalten darstellt. Der Epson-Drucker kehrt nach der Ausgabe der spezifizierten Anzahl Musterbytes wieder in den Textmodus zurück. (H. Eyssele/hm)

```
 programm : centronics      c900 cbd7
-----------------------------------
c900 : a9 5a a0 c9 8d 1a 03 8c  06
c908 : 1b 03 a9 91 a0 c9 8d 1c  08
c910 : 03 8c 1d 03 a9 ad a0 c9  1f
c918 : 8d 1e 03 8c 1f 03 a9 c8  49
c920 : a0 c9 8d 20 03 8c 21 03  2b
c928 : a9 e3 a0 c9 8d 26 03 8c  53
c930 : 27 03 a9 ff 8d 03 dd ad  07
c938 : 02 dd 09 04 8d 02 dd 60  0d
c940 : 48 a9 10 2c 0d dd f0 fb  62
c948 : 68 8d 01 dd ad 00 dd 09  d7
c950 : 04 8d 00 dd 29 fb 8d 00  7f
c958 : dd 60 a6 b8 f0 05 20 0f  fc
c960 : f3 d0 03 4c fe f6 a6 98  79
c968 : e0 0a 90 03 4c fb f6 e6  20
c970 : 98 a5 b8 9d 59 02 a5 b9  6c
c978 : 09 60 9d 6d 02 a5 ba 9d  3a
c980 : 63 02 c9 04 f0 04 c9 10  4e
c988 : 90 02 18 60 c9 00 4c 77  e8
c990 : f3 20 14 f3 f0 02 18 60  57
c998 : 20 1f f3 8a 48 a5 ba c9  c6
c9a0 : 10 b0 07 c9 04 f0 03 4c  70
c9a8 : 9d f2 4c f1 f2 20 0f f3  64
c9b0 : f0 03 4c 01 f7 20 1f f3  3a
c9b8 : a5 ba c9 04 f0 04 c9 10  24
c9c0 : 90 03 4c 0a f7 4c 19 f2  52
c9c8 : 20 0f f3 f0 03 4c 01 f7  11
c9d0 : 20 1f f3 a5 ba c9 04 f0  1d
c9d8 : 04 c9 10 90 03 4c 75 f2  25
c9e0 : 4c 5b f2 48 85 9e a5 9a  b9
c9e8 : c9 10 b0 07 c9 04 f0 03  4d
c9f0 : 4c cd f1 98 48 8a 48 a5  f7
c9f8 : 9e a4 9a c0 10 d0 06 20  87
ca00 : 40 c9 18 90 1f c0 11 d0  1b
ca08 : 06 20 5c ca 18 90 15 c0  6b
ca10 : 04 f0 04 c0 12 d0 06 20  a5
ca18 : 2b ca 18 90 07 c0 13 d0  25
ca20 : 03 20 44 ca 68 aa 68 a8  6d
ca28 : 68 18 60 c9 41 90 12 c9  62
ca30 : 5f b0 04 09 20 d0 0a c9  4e
ca38 : c1 90 06 c9 de b0 02 29  ca
ca40 : 7f 4c 40 c9 c9 ff f0 18  bf
ca48 : c9 60 b0 03 4c 40 c9 e9  90
ca50 : 40 10 02 e9 40 a0 d0 84  ab
ca58 : 06 4c c4 ca c9 ff d0 06  fb
ca60 : a2 5e a0 d0 d0 5b 48 a4  c6
ca68 : b9 c0 ff d0 02 e6 b9 29  2c
ca70 : 7f c9 20 90 2c a8 a5 b9  00
ca78 : 29 01 f0 10 68 c9 a0 90  d8
ca80 : 04 c9 c0 90 03 4c 2b ca  80
ca88 : e9 40 d0 0e 68 c9 60 b0  3f
ca90 : 03 4c 40 c9 e9 40 10 02  e7
ca98 : e9 40 a0 d0 84 06 4c c4  17
caa0 : ca 68 24 0f 30 03 4c 40  56
caa8 : c9 18 69 40 30 02 69 40  19
cab0 : aa a5 b9 29 02 d0 71 a0  6e
cab8 : d0 a5 b9 29 01 f0 02 a0  cf
cac0 : d8 84 06 8a a8 a9 00 a2  ca
cac8 : 07 9d c0 02 ca 10 fa 98  58
cad0 : 4a 4a 4a 4a 4a 18 65 06  22
cad8 : 85 06 98 0a 0a 0a 85 05  d9
cae0 : a9 01 85 03 78 a5 01 29  d7
cae8 : fb 85 01 a0 07 b1 05 85  17
caf0 : 02 a2 07 06 02 90 08 bd  06
caf8 : c0 02 05 03 9d c0 02 ca  d9
cb00 : 10 f1 06 03 88 10 e6 a5  db
cb08 : 01 09 04 85 01 58 a2 00  9d
cb10 : bd d2 cb 20 40 c9 e8 e0  e5
cb18 : 05 d0 f5 a2 07 bd c0 02  bc
cb20 : 20 40 c9 ca 10 f7 18 60  0e
cb28 : e0 d5 90 01 ca 8a 29 7f  dc
cb30 : c9 50 90 02 e9 03 c9 45  ee
cb38 : 90 02 e9 03 c9 41 90 02  91
cb40 : e9 21 c9 1c 90 02 e9 08  81
cb48 : c9 11 90 02 e9 0b 38 e9  aa
cb50 : 04 aa a9 3c 20 40 c9 a0  08
cb58 : ff ca f0 08 c8 b9 7a cb  d5
cb60 : 10 fa 30 f5 c8 b9 7a cb  94
cb68 : 30 06 20 40 c9 18 90 f4  35
cb70 : 29 7f 20 40 c9 a9 3e 4c  e4
cb78 : 40 c9 57 48 d4 43 52 c4  b6
cb80 : 52 4f ce 48 4f cd 52 45  6e
```

Listing 1. Der Softwareteil der Eyssele-Schnittstelle. Beachten Sie bitte die Eingabehinweise zum MSE.

```
cb88 : c4 43 52 d2 47 52 ce 42  a3
cb90 : 4c d5 4f 52 c7 46 b1 46  e7
cb98 : b3 46 b5 46 b7 46 b2 46  a9
cba0 : b4 46 b6 46 b8 42 4c cb  54
cba8 : 43 52 d5 52 4f c6 43 4c  a5
cbb0 : d2 42 52 ce 48 52 d4 47  0b
cbb8 : 52 b1 47 52 b2 48 47 ce  27
cbc0 : 48 42 cc 47 52 b3 50 55  f4
cbc8 : d2 43 52 cc 59 45 cc 43  e3
cbd0 : 59 ce 1b 2a 04 08 00 ff  1d
```

Listing 1. Der Softwareteil der Eyssele-Schnittstelle. Beachten Sie bitte die Eingabehinweise zum MSE.

```
10 REM 5 X CURSOR DOWN
20 PRINT "<CRD><CRD><CRD><CRD><CRD>"
30 REM 5 X CURSOR LEFT
40 PRINT "<CRL><CRL><CRL><CRL><CRL>"
50 REM BEIDES GEMISCHT
60 PRINT "<CRD><CRL><CRD><CRL><CRD>"
70 REM ROT SCHREIBEN
80 PRINT "<RED>"
90 PRINT SPC(10)"EYSSELE SCHNITTSTELLE"
100 REM SCHWARZ SCHREIBEN
110 PRINT "<BLK>"
120 PRINT"GERAET 16   = DIREKTMODUS"
130 PRINT"GERAET 18,4 = TEXT"
140 PRINT"GERAET 19   = GRAFIK"
150 PRINT"GERAET 17   = LIST"
160 PRINT"SEK-ADR. 0  = GROSS/GRAFIK"
170 PRINT"SEK-ADR. 1  = GROSS/KLEIN"
180 REM MIT ERKLAERTEN STEUERZEICHEN:
190 PRINT"SEK-ADR. 2  = GROSS/GRAFIK"
200 PRINT"SEK-ADR. 3  = GROSS/KLEIN"
210 REM "┌────────────────────────┐"
220 REM "| AUCH GRAFIKZEICHEN     |"
230 REM "|        WERDEN GEDRUCKT! |"
240 REM "└────────────────────────┘"
READY.
```

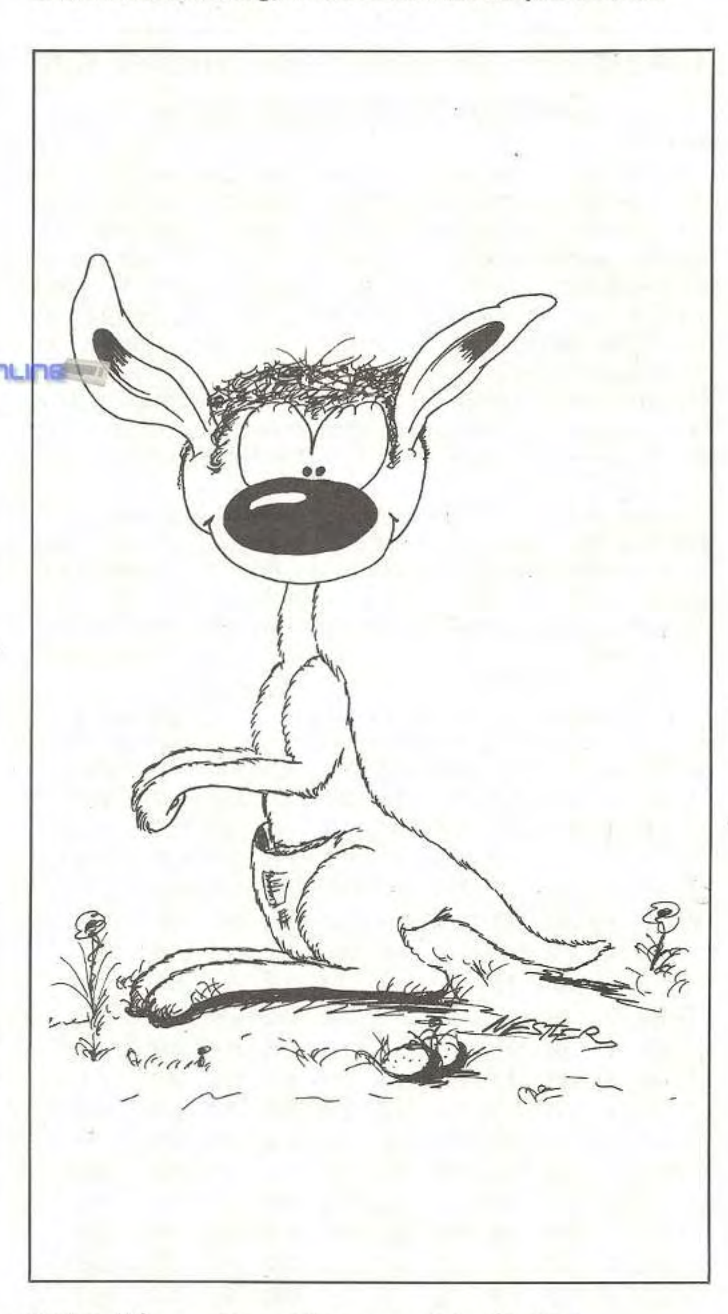
Bild 1. Steuerzeichen werden von dieser Schnittstelle genauso bewältigt wie Grafikzeichen

```
10 REM 5 X CURSOR DOWN
20 PRINT "▯▯▯▯▯"
30 REM 5 X CURSOR LEFT
40 PRINT "▮▮▮▮▮"
50 REM BEIDES GEMISCHT
60 PRINT "▯▮▯▮▯"
70 REM ROT SCHREIBEN
80 PRINT "▣"
90 PRINT SPC(10)"EYSSELE SCHNITTSTELLE"
100 REM SCHWARZ SCHREIBEN
110 PRINT "■"
120 PRINT"GERAET 16   = DIREKTMODUS"
130 PRINT"GERAET 18,4 = TEXT"
140 PRINT"GERAET 19   = GRAFIK"
150 PRINT"GERAET 17   = LIST"
160 PRINT"SEK-ADR. 0  = GROSS/GRAFIK"
170 PRINT"SEK-ADR. 1  = GROSS/KLEIN"
180 REM MIT ERKLAERTEN STEUERZEICHEN:
190 PRINT"SEK-ADR. 2  = GROSS/GRAFIK"
200 PRINT"SEK-ADR. 3  = GROSS/KLEIN"
210 REM "┌────────────────────────┐"
220 REM "| AUCH GRAFIKZEICHEN     |"
230 REM "|        WERDEN GEDRUCKT! |"
240 REM "└────────────────────────┘"
READY.
```

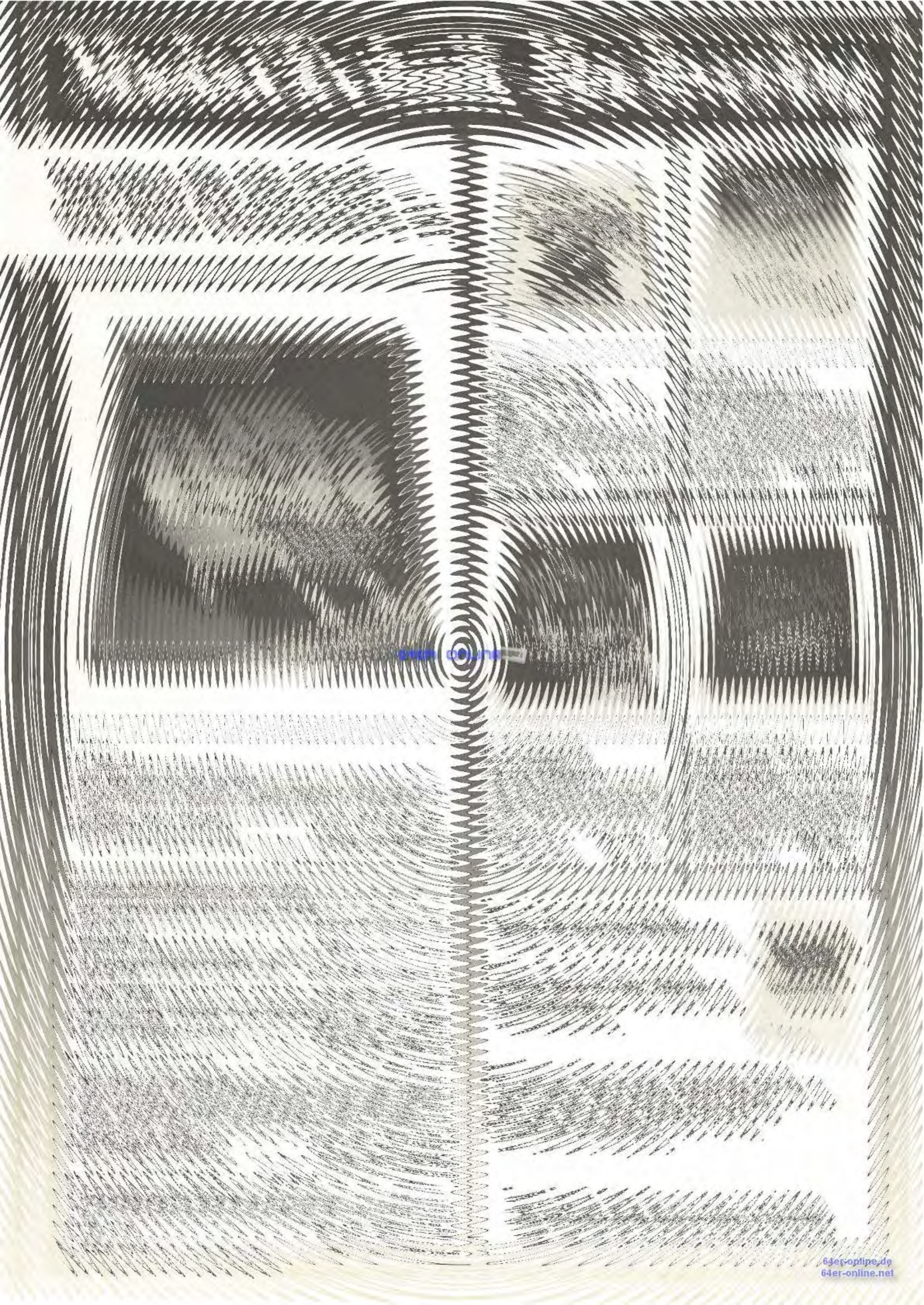
```
Druck mit OPEN 17,17,0:

100 POKE 53280,6: POKE 53281,6
110 PRINT CHR$(14)"▯▯▯▯▯▯▯▯▯";
120 PRINT CHR$(14)"▮▮▮▮▮▮▮▮▮";
130 PRINT CHR$(14)"▣*****************************"
140 PRINT "▮▮▮▮▮▮▮▮▮";
150 PRINT "*                         *"
160 PRINT "▮▮▮▮▮▮▮▮▮";
170 PRINT "*:: --64 |REIBERPROGRAMM ▣ *"
180 PRINT "▮▮▮▮▮▮▮▮▮";
190 PRINT "*                         *"
200 PRINT "▮▮▮▮▮▮▮▮▮";
210 PRINT "* ▣  FUER DEN ♠NSCHLUSS  ▣ *"
220 PRINT "▮▮▮▮▮▮▮▮▮";
230 PRINT "* ▣VON -‾/|_┌/\-♥ -RUCKER▣ *"
240 PRINT "▮▮▮▮▮▮▮▮▮";
250 PRINT "* ▣    AM ,♥‾_ - ┐ORT    ▣ *"
260 PRINT "▮▮▮▮▮▮▮▮▮";
270 PRINT "*                         *"
280 PRINT "▮▮▮▮▮▮▮▮▮";
290 PRINT "****************************▣"
```

Bild 3. Musterlisting, erstellt auf einem Epson FX-80



◄ **Bild 2. Steuerzeichen können auch in Klartext übersetzt werden**



Epson-Support

Epson-Support läßt die Herzen eines jeden FX/RX-80-Besitzers höher schlagen. Mit einfachen Befehlen und kurzen Programmen können Sie jetzt komplizierte Grafiken auf Ihrem Drucker erstellen. Und das mit einer Auflösung, die fast siebenmal so hoch ist wie die des C 64, nämlich bis zu 800 mal 552 Punkten.

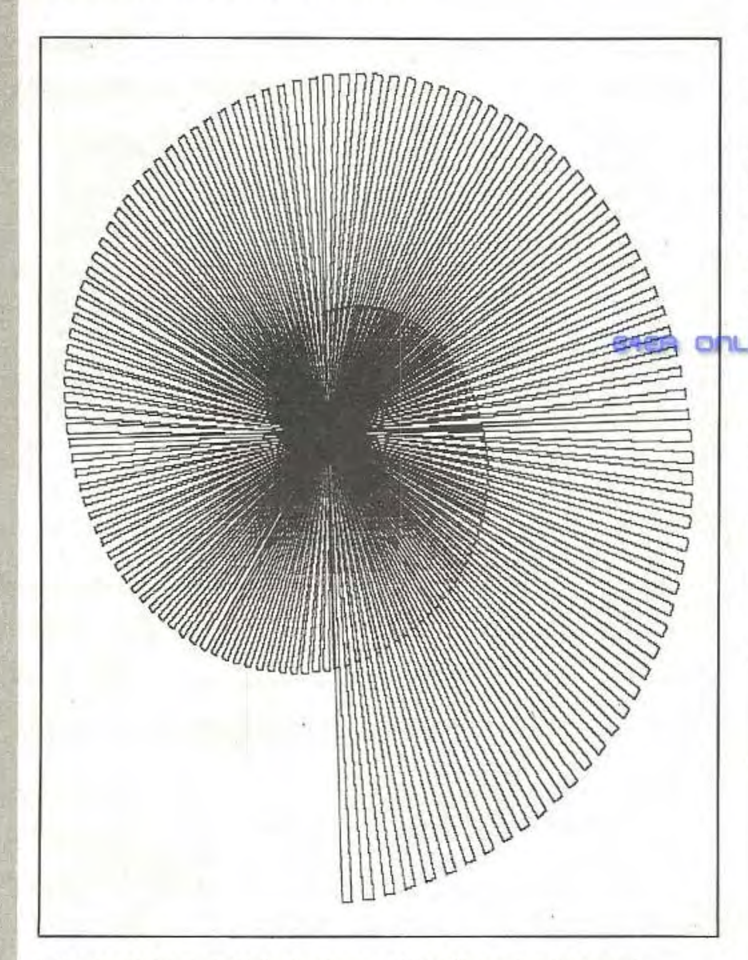


Bild 1. Listing 1 liefert dieses Bild (hier verkleinert)

Epson-Support ist eine Basic-Erweiterung und belegt den Speicherbereich ab $C000 (49152). Der Programmstart erfolgt durch SYS 49152; jetzt stehen einige leistungsfähige Befehle zur Verfügung, um Grafiken zu erstellen und auf den Epson FX/RX-Druckern auszugeben. Die Grafiken liegen im Speicher des Commodore 64 ab $2000 (8192). Wegen des enormen Speicherplatzbedarfs wird auch das RAM unter dem Basic, dem Betriebssystem und unter dem Zeichen-ROM verwendet. Die Bitmap wird über Koordinaten adressiert, dabei liegt der Punkt (0,0) in der linken Ecke oben wie bei den normalen HiRes-Grafiken auf dem Bildschirm. Die X-Werte laufen von 0 bis 799, die Y-Werte von 0 bis 551, so daß 800 x 552 = 441600 Dots adressiert werden können. Die Bitmap ist so organisiert, daß der Ausdruck relativ schnell geht.

Die Ausgabe mit dem Drucker läuft über ein User-Port-Kabel, die nötige Treibersoftware ist im Epson-Support schon enthalten. Der Ausdruck erfolgt in der sogenannten doppelten Zeichendichte, also mit 960 Punkten pro Zeile, aus Speichermangel können aber nur 800 Punkte pro Zeile verwendet werden. Listing 1 und Listing 2 sind zwei Beispielprogramme. Als Ergebnis liefern sie Bild 1 und Bild 2. Bevor Sie diese Beispiele probieren, muß das Listing 3 mit dem MSE (siehe Seite 7) eingegeben werden und mit SYS49152 initialisiert werden.

Epson-Support Bedienungsanleitung

Das Programm Epson-Support wird absolut geladen mit LOAD"EPSON-SUPPORT",8,1 und durch SYS 49152 gestartet. Dem Basic-Programmierer stehen dann einige neue Befehle zur Verfügung, um die grafischen Fähigkeiten des Epson-Druckers besser auszunutzen, denn im Speicher des C 64 wird eine Grafik-Bitmap von 800 x 552 Punkten verwaltet. Die Auflösung ist also wesentlich höher als bei normalen Bildschirm-Grafiken.

Die einzelnen Befehle werden im folgenden erläutert:

&CLEAR

Damit wird die Grafik-Bitmap gelöscht. Dieser Befehl sollte unbedingt am Anfang eines jeden Programms stehen, sonst stehen in der Bitmap wirre Muster.

&DOT X,Y

Dieser Befehl setzt einen Punkt an der Position (X,Y). Der X-Wert liegt zwischen 0 (links) und 799 (rechts), der Y-Wert zwischen 0 (oben) und 551 (unten). Für X und Y kann ein beliebiger numerischer Basic-Ausdruck stehen.

&RESETDOT X,Y

Dieser Befehl löscht einen Punkt an der Position (X,Y). Der X-Wert liegt zwischen 0 (links) und 799 (rechts), der Y-Wert zwischen 0 (oben) und 551 (unten). Für X und Y kann ein beliebiger numerischer Basic-Ausdruck stehen.

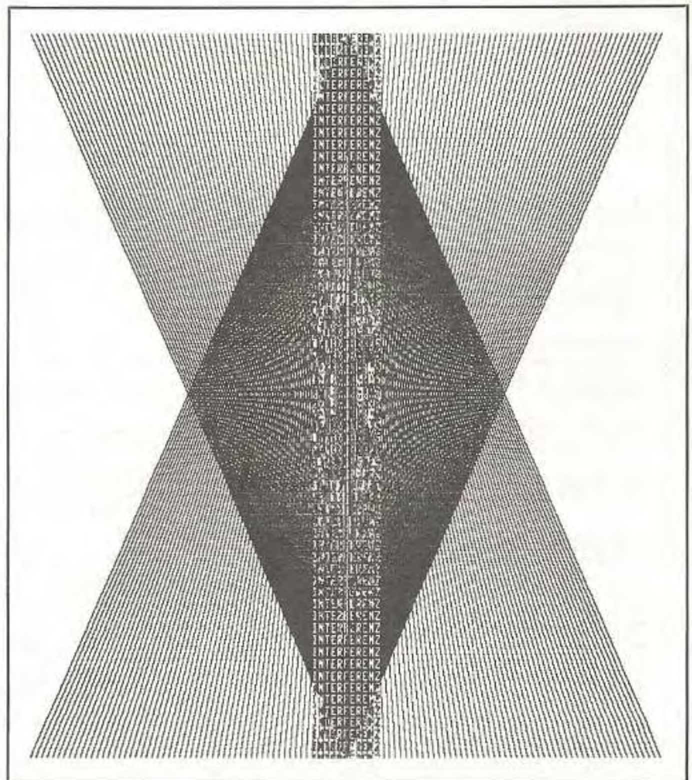


Bild 2. Listing 2 erzeugt dieses Bild (hier verkleinert)

&INVDOT X,Y

Dieser Befehl invertiert einen Punkt an der Position (X,Y). Der X-Wert liegt zwischen 0 (links) und 799 (rechts), der Y-Wert zwischen 0 (oben) und 551 (unten). Für X und Y kann ein beliebiger numerischer Basic-Ausdruck stehen.

&LINE X1,Y1,X2,Y2

Dieser Befehl zieht eine Linie zwischen den Punkten (X1,Y1) und (X2,Y2). Der Wertebereich der Koordinaten ist der gleiche wie bei den Dot-Befehlen. Für die Parameter können beliebige Basic-Ausdrücke stehen.

&RESETLINE X1,Y1,X2,Y2

Dieser Befehl löscht eine Linie zwischen den Punkten (X1,Y1) und (X2,Y2). Der Wertbereich der Koordination ist der gleiche wie bei den Dot-Befehlen. Für die Parameter können beliebige Basic-Ausdrücke stehen.

&INVLINE X1,Y1,X2,Y2

Dieser Befehl invertiert die Punkte einer Linie zwischen den Punkten (X1,Y1) und (X2,Y2). Der Wertebereich der Koordinaten ist wiederum der gleiche wie bei den Dot-Befehlen. Für die Parameter können beliebige Basic-Ausdrücke stehen.

&SCANLINE X1,Y1,X2,Y2

Dieser Befehl setzt eine Linie zwischen den Punkten (X1,Y1) und (X2,Y2). Falls jedoch auf einen schon gesetzten Punkt auf dem Linienzug getroffen wird, so wird nicht mehr weitergezeichnet. Der Wertebereich der Koordinaten ist der gleiche wie bei den Dot-Befehlen. Für die Parameter können beliebige Basic-Ausdrücke stehen.

&PRINT

Durch diesen Befehl wird die HiRes-Bitmap auf dem Drucker ausgegeben. Das Programm ist für Epson FX/RX-80-Drucker ausgelegt. Bei Verwendung eines anderen Druckers müssen einige Stellen im Assemblerlisting (Listing 4) beim &PRINT-Befehl geändert werden. Siehe dazu auch die Kommentare im Listing bei diesem Befehl.

&EXIT

Führt zum Verlassen von Epson-Support und bewirkt danach einen Basic-Neustart. Basic-Programme sollten vorher unbedingt abgespeichert sein, weil sie sonst gelöscht werden.

&TEXT X,Y, "blablabla..."

Dieser Befehl setzt den Text, welcher in Anführungszeichen steht, in die Bitmap an der Position (X,Y). Diese Position gibt die Koordinaten der linken unteren Ecke des ersten Buchstabens an.

&RESETTEXT, X,Y, "blablabla..."

Dieser Befehl löscht den Text, welcher in Anführungszeichen steht, in die Bitmap an der Position (X,Y). Diese Position gibt die Koordinaten der linken unteren Ecke des ersten Buchstabens an.

&INVTEXT X,Y, "blablabla..."

Dieser Befehl invertiert den Text, welcher in Anführungszeichen steht, in die Bitmap an der Position (X,Y). Diese Position gibt die Koordinaten der linken unteren Ecke des ersten Buchstabens an.

(Alexander Del Pino/gk)

```
100 REM                                        <162>
110 REM DEMO1 - SCHNECKE                       <158>
120 REM                                        <182>
130 &CLEAR                                     <025>
140 D=↑/120:DX=400:DY=276                      <233>
150 SX=400/240:SY=276/240                      <050>
160 FOR I=0 TO 4*↑STEP↑/60                     <096>
170 X1=SIN(I)*DX:Y1=COS(I)*DY                  <080>
180 X2=SIN(I+D)*DX:Y2=COS(I+D)*DY              <051>
190 X1=X1+400:X2=X2+400:Y1=Y1+276:Y2=Y2+27
    6                                          <173>
200 &LINE400,276,X1,Y1                         <066>
210 &LINEX1,Y1,X2,Y2                           <073>
220 &LINEX2,Y2,400,276                         <185>
230 DX=DX-SX:DY=DY-SY                          <194>
240 PRINT".";                                  <091>
250 NEXT                                       <004>
260 &PRINT:END                                 <029>
```

© 64'er

Listing 1. Demo-Programm »Schnecke«

```
100 REM                                        <162>
110 REM DEMO2 - INTERFERENZ                    <009>
120 REM                                        <182>
130 &CLEAR                                     <025>
140 FOR I=0 TO 799 STEP 7                      <219>
150 PRINT".";                                  <001>
160 &LINE400,0,I,552                           <076>
170 &LINE400,552,I,0                           <041>
180 NEXT                                       <190>
190 FOR I=0 TO 552 STEP 9                      <010>
200 PRINT".";                                  <051>
210 &INVTEXT356,I,"INTERFERENZ"                <071>
220 NEXT                                       <230>
230 &PRINT:END                                 <253>
```

© 64'er

Listing 2. Demo-Programm »Interferrenz«

```
programm : epson support    c000 c54e
--------------------------------------
c000 : a9 1d a0 c0 8d 08 03 8c  b6
c008 : 09 03 a9 fe a0 1f 85 33  5c
c010 : 84 34 85 37 84 38 a9 a4  f1
c018 : a0 c4 4c 1e ab 20 73 00  7b
c020 : c9 26 f0 06 20 79 00 4c  60
c028 : e7 a7 a2 ff a0 00 8c 62  8c
c030 : 03 e8 c8 bd d7 c4 f0 17  27
c038 : d1 7a f0 f5 e8 bd d7 c4  a6
c040 : d0 fa ee 62 03 a0 00 bd  46
c048 : d8 c4 d0 e5 4c 08 af 88  48
c050 : 18 98 65 7a 85 7a 90 02  cf
c058 : e6 7b a9 a7 48 a9 e9 48  65
c060 : ad 62 03 0a aa bd 31 c5  29
c068 : 48 bd 30 c5 48 4c 73 00  08
c070 : a9 00 a0 20 85 fb 84 fc  8a
c078 : 78 a9 34 85 01 a0 00 98  c9
c080 : 91 fb c8 d0 fb e6 fc a6  93
c088 : fc e0 c0 d0 f3 a2 c8 86  c3
c090 : fc 91 fb c8 d0 fb e6 fc  ef
c098 : d0 f7 a9 37 85 01 58 60  38
c0a0 : a9 00 8d 62 03 20 4e c3  eb
c0a8 : 8d 3c 03 8c 3d 03 20 4b  a9
c0b0 : c3 c0 02 90 05 c9 28 90  c6
c0b8 : 01 60 8d 3e 03 8c 3f 03  ac
c0c0 : a0 03 18 4e 3f 03 6e 3e  f4
c0c8 : 03 88 d0 f6 18 ad 3e 03  10
c0d0 : 69 20 85 fc ad 3e 03 85  2e
c0d8 : fb 0a 65 fc 85 fc a9 00  b8
c0e0 : 06 fb 2a 06 fb 2a 06 fb  50
c0e8 : 2a 06 fb 2a 06 fb 2a 65  0d
c0f0 : fc 85 fc a5 14 29 07 aa  9f
c0f8 : bd 88 c4 8d 63 03 ad 3c  5a
c100 : 03 ac 3d 03 c0 03 90 05  7a
c108 : c9 20 90 01 60 65 fb 85  52
c110 : fb 98 65 fc c9 c0 90 02  39
c118 : 69 07 85 fc 78 a9 34 85  b6
c120 : 01 a0 00 2c 62 03 10 1c  ae
c128 : 50 10 ad 63 03 a2 00 31  00
c130 : fb f0 01 ca 8e 62 03 4c  de
c138 : 59 c1 ad 63 03 51 fb 91  17
c140 : fb 4c 59 c1 50 0c ad 63  d3
c148 : 03 49 ff 31 fb 91 fb 4c  eb
c150 : 59 c1 ad 63 03 11 fb 91  2d
c158 : fb a9 37 85 01 58 60 78  ec
c160 : a9 ff 8d 03 dd ad 02 dd  dc
c168 : 09 04 8d 02 dd ad 00 dd  1e
c170 : 09 04 8d 00 dd a9 10 8d  65
c178 : 0d dd ad 0d dd 58 a2 05  b6
c180 : bd 90 c4 20 f1 c1 ca 10  33
c188 : f7 a9 00 a0 20 85 fb 84  8f
c190 : fc a9 00 85 fd a2 00 bd  82
c198 : 96 c4 20 f1 c1 e8 e0 0e  da
c1a0 : d0 f5 a2 00 20 c7 c1 e8  2d
c1a8 : d0 fa 20 c7 c1 e8 d0 fa  93
c1b0 : 20 c7 c1 e8 d0 fa 20 c7  36
c1b8 : c1 e8 e0 20 d0 f8 e6 fd  96
c1c0 : a5 fd c9 45 d0 cf 60 78  7d
c1c8 : a9 34 85 01 18 a5 fb 69  7e
c1d0 : 00 85 fe a5 fc 69 08 85  4d
c1d8 : ff a0 00 a5 fc c9 c0 b0  5f
c1e0 : 03 b1 fb 2c b1 fe e6 fb  e7
c1e8 : d0 02 e6 fc a0 37 84 01  ea
c1f0 : 58 8d 01 dd ad 00 dd 29  af
c1f8 : fb 8d 00 dd 09 04 8d 00  5d
c200 : dd ad 0d dd 29 10 f0 f9  7e
c208 : 18 60 a9 00 8d 62 03 a9  06
c210 : 00 8d 6b 03 20 4e c3 8d  b1
c218 : 50 03 8c 51 03 20 4b c3  1d
c220 : 8d 52 03 8c 53 03 20 4b  8d
c228 : c3 8d 54 03 8c 55 03 20  e7
c230 : 4b c3 8d 56 03 8c 57 03  83
c238 : a2 01 a0 00 38 ad 54 03  cb
c240 : ed 50 03 8d 5c 03 ad 55  07
c248 : 03 ed 51 03 8d 5d 03 10  e7
c250 : 18 ca ca 88 49 ff 8d 5d  16
c258 : 03 ad 5c 03 49 ff 8d 5c  2d
c260 : 03 ee 5c 03 d0 03 ee 5d  ed
c268 : 03 8e 58 03 8c 59 03 a2  0e
c270 : 01 a0 00 38 ad 56 03 ed  3e
c278 : 52 03 8d 5e 03 ad 57 03  7c
c280 : ed 53 03 8d 5f 03 10 18  08
c288 : ca ca 88 49 ff 8d 5f 03  f2
c290 : ad 5e 03 49 ff 8d 5e 03  42
c298 : ee 5e 03 d0 03 ee 5f 03  bb
c2a0 : 8e 5a 03 8c 5b 03 a0 00  fe
c2a8 : ad 5c 03 0d 5d 03 d0 01  19
c2b0 : 88 8c 60 03 8c 61 03 ad  32
c2b8 : 50 03 ac 51 03 8d 3c 03  73
c2c0 : 8c 3d 03 ad 52 03 ac 53  f8
```

Listing 3. Epson-Support muß mit dem MSE eingegeben werden. Sie finden den MSE in diesem Heft auf Seite 8.

```
c2c8 : 03 85 14 84 15 20 5f c4  7d
c2d0 : 20 b1 c0 ad 50 03 cd 54  ac
c2d8 : 03 d0 19 ad 51 03 cd 55  4e
c2e0 : 03 d0 11 ad 52 03 cd 56  66
c2e8 : 03 d0 09 ad 53 03 cd 57  7f
c2f0 : 03 d0 01 60 2c 61 03 10  a2
c2f8 : 29 18 ad 52 03 6d 5a 03  ee
c300 : 8d 52 03 ad 53 03 6d 5b  e7
c308 : 03 8d 53 03 18 ad 60 03  7e
c310 : 6d 5c 03 8d 60 03 ad 61  b5
c318 : 03 6d 5d 03 8d 61 03 4c  12
c320 : b7 c2 18 ad 50 03 6d 58  78
c328 : 03 8d 50 03 ad 51 03 6d  b3
c330 : 59 03 8d 51 03 38 ad 60  02
c338 : 03 ed 5e 03 8d 60 03 ad  6d
c340 : 61 03 ed 5f 03 8d 61 03  b2
c348 : 4c b7 c2 20 fd ae 20 8a  0f
c350 : ad 20 f7 b7 a5 14 a4 15  ba
c358 : 60 a9 40 4c a2 c0 a9 80  fe
c360 : 4c a2 c0 a9 40 4c 0c c2  7f
c368 : a9 80 4c 0c c2 a0 a7 a9  09
c370 : e4 8d 08 03 8c 09 03 38  0b
c378 : 20 99 ff 86 37 84 38 4c  47
c380 : 9a e3 a9 80 2c a9 40 2c  f0
c388 : a9 00 8d 62 03 20 4e c3  d3
c390 : 8d 50 03 8c 51 03 20 4b  dc
c398 : c3 8d 52 03 8c 53 03 20  c7
c3a0 : 40 c4 d0 01 60 8d 6a 03  b9
c3a8 : 8c 69 03 ac 69 03 b1 64  7d
c3b0 : c9 20 90 0b c9 40 90 09  02
c3b8 : c9 60 b0 03 29 1f 2c a9  cd
c3c0 : 20 85 02 a9 1a 06 02 2a  87
c3c8 : 06 02 2a 06 02 2a 85 03  a8
c3d0 : 78 a9 31 85 01 a0 07 b1  ae
c3d8 : 02 99 00 02 88 10 f8 a9  27
c3e0 : 37 85 01 58 a9 00 8d 67  c5
c3e8 : 03 a9 07 8d 68 03 ae 68  5e
c3f0 : 03 bd 00 02 ae 67 03 3d  bf
c3f8 : 88 c4 f0 1e ad 50 03 ac  a5
c400 : 51 03 8d 3c 03 8c 3d 03  4d
c408 : ac 53 03 ad 68 03 18 6d  ae
c410 : 52 03 85 14 90 01 c8 20  3c
c418 : b1 c0 ce 68 03 10 cf ee  b8
c420 : 50 03 d0 03 ee 51 03 ee  ea
c428 : 67 03 ae 67 03 e0 08 d0  a2
c430 : b8 ee 69 03 ad 69 03 cd  e8
c438 : 6a 03 d0 01 60 4c ab c3  17
c440 : 20 fd ae 20 9e ad 20 a3  2e
c448 : b6 a0 00 b1 64 85 63 c8  16
c450 : b1 64 aa c8 b1 64 85 65  16
c458 : 86 64 a0 00 a5 63 60 2c  88
c460 : 6b 03 30 01 60 a2 f0 8e  75
c468 : 62 03 20 b1 c0 ad 52 03  53
c470 : ac 53 03 85 14 2c 62 03  6a
c478 : 10 03 68 68 60 a2 00 8e  69
c480 : 62 03 60 a9 ff 4c 11 c2  dd
c488 : 80 40 20 10 08 04 02 01  dd
c490 : 0d 08 41 1b 38 1b 0a 0d  f4
c498 : 20 20 20 20 20 20 20 1b  8e
c4a0 : 2a 01 20 03 45 50 53 4f  76
c4a8 : 4e 2d 53 55 50 50 4f 52  76
c4b0 : 54 20 20 56 31 2e 37 20  89
c4b8 : 20 2a 2a 2a 0d 20 20 31  72
c4c0 : 39 38 35 20 56 4f 4e 20  c0
c4c8 : 41 4c 45 58 20 44 45 4c  5e
c4d0 : 20 50 49 4e 4f 0d 00 43  18
c4d8 : 4c 45 41 52 00 44 4f 54  6a
c4e0 : 00 52 45 53 45 54 44 4f  6c
c4e8 : 54 00 49 4e 56 44 4f 54  c6
c4f0 : 00 4c 49 4e 45 00 52 45  5b
c4f8 : 53 45 54 4c 49 4e 45 00  a9
c500 : 49 4e 56 4c 49 4e 45 00  ac
c508 : 99 00 45 58 49 54 00 54  de
c510 : 45 58 54 00 49 4e 56 54  a0
c518 : 45 58 54 00 52 45 53 45  c6
c520 : 54 54 45 58 54 00 53 43  14
c528 : 41 4e 4c 49 4e 45 00 00  dc
c530 : 6f c0 9f c0 58 c3 5d c3  a0
c538 : 09 c2 62 c3 67 c3 5e c1  45
c540 : 6c c3 87 c3 81 c3 84 c3  b8
c548 : 82 c4 00 00 c9 00 00 ff  c9
```

Listing 3. Epson-Support muß mit dem MSE eingegeben werden. Sie finden den MSE in diesem Heft auf Seite 8. (Schluß)

```
100             .BA $C000 ;PC SETZEN
105             ;************************************
110             ; E P S O N - S U P P O R T
115             ;
120             ;VERSION 1.7    1984 BY A.DEL PINO
125             ;
130             ;************************************
131             ;
132             ;VARIABLEN & SYSTEMADRESSEN
133             ;
135 HELP        = $033C
140 YVALUE      = $14
145 XVALUE      = $033C
150 X1          = $0350
155 Y1          = X1+2
160 X2          = X1+4
165 Y2          = X1+6
170 XSTEP       = X1+8
175 YSTEP       = X1+10
180 XDIFF       = X1+12
185 YDIFF       = X1+14
190 SUM         = X1+16
195 XMAX        = 800
200 YMAX        = 552
205 DISABLE     = $34 ;ROM AUSBLENDEN
210 ENABLE      = 55 ;ROM EINBLENDEN
215 FLAG        = X1+18
220 BITPOS      = X1+19
225 FLAG1       = X1+23
230 FLAG2       = X1+24
235 FLAG3       = X1+25
240 LENGH       = X1+26
245 LINETYP     = X1+27
246             ;
247             ;PROGRAMMSTART, ZEIGER VERBIEGEN
248             ;
250             LDA #<ESUP
255             LDY #>ESUP
260             STA $0308
265             STY $0309
270             LDA #<8190
275             LDY #>8190
280             STA $33
285             STY $34
290             STA $37
295             STY $38
296             ;
297             ;PROMPT AUSGEBEN
298             ;
300             LDA #<PROMPT
305             LDY #>PROMPT
310             JMP $AB1E
311             ;
312             ;BASIC-TOKEN ERKENNEN
313             ;
315 ESUP        JSR $73
320             CMP #'&'
325             BEQ SUPPORT
330             JSR $79
335             JMP $A7E7
340 SUPPORT     LDX #$FF
345             LDY #0
350             STY FLAG
355 SLOOP       INX
360             INY
365             LDA §STMTAB,X
370             BEQ NEWSTMT
375             CMP ($7A),Y
380             BEQ SLOOP
385 NXT         INX
390             LDA §STMTAB,X
395             BNE NXT
400             INC FLAG
405             LDY #0
410             LDA §STMTAB+1,X
415             BNE SLOOP
420             JMP $AF08
425 NEWSTMT     DEY
430             CLC
435             TYA
440             ADC $7A
445             STA $7A
450             BCC L1
455             INC $7B
456             ;
457             ;RETURN-ADRESSE AUF DEN STACK
458             ;
460 L1          LDA #$A7
465             PHA
470             LDA #$E9
475             PHA
480             LDA FLAG
485             ASL
490             TAX
495             LDA §ENTRY+1,X
496             ;
497             ;ADRESSE DES NEUEN BEFEHLS AUF DEN STACK
500             PHA
505             LDA §ENTRY,X
510             PHA
515             JMP $73
516             ;
517             ;HIER SIND DIE NEUEN BEFEHLE...
518             ;
520             ;**************************
525             ;
530             ; &CLEAR
535             ;
540             ;**************************
545 CLEAR       LDA #<8192
550             LDY #>8192
555             STA $FB
560             STY $FC
565             SEI
570             LDA #DISABLE
575             STA 1
580             LDY #0
585             TYA
590 CL1         STA ($FB),Y
595             INY
600             BNE CL1
605             INC $FC
610             LDX $FC
615             CPX #$C0
620             BNE CL1
625             LDX #$CB
630             STX $FC
635 CL2         STA ($FB),Y
640             INY
645             BNE CL2
650             INC $FC
655             BNE CL2
660             LDA #ENABLE
665             STA 1
670             CLI
675             RTS
680             ;**************************
685             ;
690             ; DOT - BEFEHLE
695             ;
700             ;**************************
705 DOT         LDA #0
710             STA FLAG
715             JSR VALUE+3
720             STA XVALUE
725             STY XVALUE+1
730             JSR VALUE
735 WORKDOT     CPY #>YMAX
740             BCC SE1
745             CMP #<YMAX
750             BCC SE1
755             RTS
756             ;
757             ;ADRESSE DES PUNKTES BERECHNEN
758             ;
760 SE1         STA §HELP+2
765             STY §HELP+3
770             LDY #3
775 SE2         CLC
780             LSR §HELP+3
785             ROR §HELP+2
790             DEY
795             BNE SE2
800             CLC
805             LDA §HELP+2
810             ADC #>8192
815             STA $FC
820             LDA §HELP+2
825             STA $FB
830             ASL
835             ADC $FC
840             STA $FC
845             LDA #0
850             ASL $FB
855             ROL
```

Listing 4. Der Source-Code zum Epson-Support

```
860            ASL $FB
865            ROL
870            ASL $FB
875            ROL
880            ASL $FB
885            ROL
890            ASL $FB
895            ROL
900            ADC $FC
905            STA $FC
910 SE4        LDA $14
915            AND #7
920            TAX
925            LDA §BIT,X
930            STA BITPOS
935            LDA XVALUE
940            LDY XVALUE+1
945            CPY #>XMAX
950            BCC SE5
955            CMP #<XMAX
960            BCC SE5
965            RTS
970 SE5        ADC $FB
975            STA $FB
980            TYA
985            ADC $FC
990            CMP #$C0
995            BCC SE6
1000            ADC #7
1005 SE6        STA $FC
1010            SEI
1011            ;
1012            ;ROM AUSBLENDEN
1013            ;
1015            LDA #DISABLE
1020            STA 1
1025            LDY #0
1030            BIT FLAG
1035            BPL DOTOX
1040            BVC INVERT
1041            ;
1042            ;PUNKT TESTEN 'BITTEST'
1043            ;
1045            LDA BITPOS
1050            LDX #0
1055            AND ($FB),Y
1060            BEQ TS1
1065            DEX
1070 TS1        STX FLAG
1075            JMP ENDDOT
1076            ;
1077            ;PUNKT INVERTIEREN
1078            ;
1080 INVERT     LDA BITPOS
1085            EOR ($FB),Y
1090            STA ($FB),Y
1095            JMP ENDDOT
1100 DOTOX      BVC SET
1101            ;
1102            ;PUNKT ZURUECKSETZEN
1103            ;
1105            LDA BITPOS
1110            EOR #$FF
1115            AND ($FB),Y
1120            STA ($FB),Y
1125            JMP ENDDOT
1126            ;
1127            ;PUNKT SETZEN
1128            ;
1130 SET        LDA BITPOS
1135            ORA ($FB),Y
1140            STA ($FB),Y
1145 ENDDOT     LDA #ENABLE
1150            STA 1
1155            CLI
1160            RTS
1165            ;**************************
1170            ;
1175            ; &PRINT
1180            ;
1185            ;**************************
1186            ;USERPORT-INTERFACE DURCH GEEIGNTES
1187            ;SETZEN DER PORTBAUSTEIN-REGISTER
1188            ;INITIALISIEREN.
                 BEI DRUCKERN AM SERIELLEN BUS MUSS
1189            ;HIER EIN FILE AUF DEN DRUCKER ERDEFFNET WERDEN.
1190 HARDCOPY   SEI
1195            LDA #$FF
1200            STA $DD03
1205            LDA $DD02
1210            ORA #4
1215            STA $DD02
1220            LDA $DD00
1225            ORA #4
1230            STA $DD00
1235            LDA #$10
1240            STA $DD0D
1245            LDA $DD0D
1250            CLI
1251            ;STEUERCODES FUER EPSON RX-80 AUSGEBEN
1255            LDX #5
1260 HC0        LDA §TABLE1,X
1265            JSR CWROUT
1270            DEX
1275            BPL HC0
1280            LDA #<8192
1285            LDY #>8192
1290            STA $FB
1295            STY $FC
1300            LDA #0
1305            STA $FD
1306            ;STEUERCODES FUER NEUE GRAFIKZEILE AUSGEBEN
1310 HC1        LDX #0
1315            LDA §TABLE2,X
1320            JSR CWROUT
1325            INX
1330            CPX #14
1335            BNE HC1+2
1336            ;EINE GRAFIKZEILE (8*800 DOTS) AUSGEBEN
1340            LDX #0
1345 HC2        JSR WROUT
1350            INX
1355            BNE HC2
1360 HC3        JSR WROUT
1365            INX
1370            BNE HC3
1375 HC4        JSR WROUT
1380            INX
1385            BNE HC4
1390 HC5        JSR WROUT
1395            INX
1400            CPX #<XMAX
1405            BNE HC5
1410            INC $FD
1415            LDA $FD
1420            CMP #69
1425            BNE HC1
1430            RTS
1431            ;ROM AUSBLENDEN
1435 WROUT      SEI
1440            LDA #DISABLE
1445            STA 1
1446            ;ADRESSE DES AUSZUGEBENDEN PUNKTES BERECHNEN
1450            CLC
1455            LDA $FB
1460            ADC #<$800
1465            STA $FE
1470            LDA $FC
1475            ADC #>$800
1480            STA $FF
1485            LDY #0
1490            LDA $FC
1495            CMP #$C0
1500            BCS WR1
1505            LDA ($FB),Y
1510            .BY $2C
1515 WR1        LDA ($FE),Y
1520            INC $FB
1525            BNE WR2
1530            INC $FC
1531            ;ROM WIEDER EINBLENDEN
1535 WR2        LDY #ENABLE
1540            STY 1
1545            CLI
1546            ;GEBE ASCII-ZEICHEN IM ACCU AUS
1547            ;BEI DRUCKERN MIT SERIELLER SCHNITTSTELLE
1548            ;MUSS HIER 'JMP $FFD2' STEHEN
1549            ;UND DANACH AUCH EIN 'RTS'
1550 CWROUT     STA $DD01
1555            LDA $DD00
1560            AND #$FB
1565            STA $DD00
1570            ORA #4
1575            STA $DD00
1580 WAIT       LDA $DD0D
1585            AND #$10
1590            BEQ WAIT
1595            CLC
1600            RTS
1605            ;**************************
1610            ;
1615            ; LINE-BEFEHLE
1620            ;
1625            ;**************************
1630 LINE       LDA #0
1635            STA FLAG
1640            LDA #0
1645            STA LINETYP
1650            JSR VALUE+3
1655            STA X1
1660            STY X1+1
1665            JSR VALUE
1670            STA Y1
1675            STY Y1+1
1680            JSR VALUE
1685            STA X2
1690            STY X2+1
1695            JSR VALUE
1700            STA Y2
1705            STY Y2+1
1710            LDX #1
1715            LDY #0
1720            SEC
1725            LDA X2
1730            SBC X1
1735            STA XDIFF
1740            LDA X2+1
1745            SBC X1+1
1750            STA XDIFF+1
1755            BPL LI1
1760            DEX
1765            DEX
1770            DEY
1775            EOR #$FF
1780            STA XDIFF+1
1785            LDA XDIFF
1790            EOR #$FF
1795            STA XDIFF
1800            INC XDIFF
1805            BNE LI1
1810            INC XDIFF+1
1815 LI1        STX XSTEP
1820            STY XSTEP+1
1825            LDX #1
1830            LDY #0
1835            SEC
1840            LDA Y2
1845            SBC Y1
1850            STA YDIFF
1855            LDA Y2+1
1860            SBC Y1+1
1865            STA YDIFF+1
1870            BPL LI2
1875            DEX
1880            DEX
1885            DEY
1890            EOR #$FF
1895            STA YDIFF+1
1900            LDA YDIFF
1905            EOR #$FF
1910            STA YDIFF
1915            INC YDIFF
1920            BNE LI2
1925            INC YDIFF+1
1930 LI2        STX YSTEP
1935            STY YSTEP+1
1940            LDY #0
```

Listing 4. Der Source-Code zum Epson-Support (Fortsetzung)

```
1945            LDA XDIFF
1950            ORA XDIFF+1
1955            BNE LI3
1960            DEY
1965 LI3        STY SUM
1970            STY SUM+1
1975 LI4        LDA X1
1980            LDY X1+1
1985            STA XVALUE
1990            STY XVALUE+1
1995            LDA Y1
2000            LDY Y1+1
2005            STA YVALUE
2010            STY YVALUE+1
2015            JSR SCANN
2020            JSR WORKDOT
2025            LDA X1
2030            CMP X2
2035            BNE LI5
2040            LDA X1+1
2045            CMP X2+1
2050            BNE LI5
2055            LDA Y1
2060            CMP Y2
2065            BNE LI5
2070            LDA Y1+1
2075            CMP Y2+1
2080            BNE LI5
2085            RTS
2090 LI5        BIT SUM+1
2095            BPL LI6
2100            CLC
2105            LDA Y1
2110            ADC YSTEP
2115            STA Y1
2120            LDA Y1+1
2125            ADC YSTEP+1
2130            STA Y1+1
2135            CLC
2140            LDA SUM
2145            ADC XDIFF
2150            STA SUM
2155            LDA SUM+1
2160            ADC XDIFF+1
2165            STA SUM+1
2170            JMP LI4
2175 LI6        CLC
2180            LDA X1
2185            ADC XSTEP
2190            STA X1
2195            LDA X1+1
2200            ADC XSTEP+1
2205            STA X1+1
2210            SEC
2215            LDA SUM
2220            SBC YDIFF
2225            STA SUM
2230            LDA SUM+1
2235            SBC YDIFF+1
2240            STA SUM+1
2245            JMP LI4
2250 VALUE      JSR $AEFD
2255            JSR $AD8A
2260            JSR $B7F7
2265            LDA $14
2270            LDY $15
2275            RTS
2276            ;FLAG SETZEN FUER MODIFIZIERTEN PUNKT-
                UND LINIEN-FUNKTIONEN
2280 RESETDOT   LDA #$40
2285            JMP DOT+2
2290 INVERTDOT  LDA #$80
2295            JMP DOT+2
2300 RESETLINE  LDA #$40
2305            JMP LINE+2
2310 INVERTLINELDA #$80
2315            JMP LINE+2
2320            ;**************************
2325            ;
2330            ; &EXIT
2335            ;
2340            ;**************************
2341            ;ZEIGER WIEDER 'GERADEBIEGEN'
2345 EXIT       LDY #$A7
2350            LDA #$E4
2355            STA $0308
2360            STY $0309
2365            SEC
2370            JSR $FF99
2375            STX $37
2380            STY $38
2385            JMP $E39A ;SYSTEM AUFRUFEN
2386            ; TEXT-BEFEHLE, FLAG SETZEN
2390 INVTEXT    LDA #$80
2395            .BY $2C
2400 RESTEXT    LDA #$40
2405            .BY $2C
2410 TEXT       LDA #0
2415            STA FLAG
2420            JSR VALUE+3
2425            STA X1
2430            STY X1+1
2435            JSR VALUE
2440            STA Y1
2445            STY Y1+1
2450            JSR STRING
2455            BNE TXT1
2460            RTS
2465 TXT1       STA LENGH
2470            STY FLAG3
2475 TXT2       LDY FLAG3
2480            LDA ($64),Y
2485            CMP #$20
2490            BCC TXT3
2495            CMP #$40
2500            BCC TXT3+2
2505            CMP #$60
2510            BCS TXT3
2515            AND #$1F
2520            .BY $2C
2525 TXT3       LDA #$20
2530            STA 2
2535            LDA #$1A
2540            ASL 2
2545            ROL
2550            ASL 2
2555            ROL
2560            ASL 2
2565            ROL
2570            STA 3
2575            SEI
2580            LDA #$31
2585            STA 1
2590            LDY #7
2595 TXT4       LDA (2),Y
2600            STA $200,Y
2605            DEY
2610            BPL TXT4
2615            LDA #ENABLE
2620            STA 1
2625            CLI
2630            LDA #0
2635            STA FLAG1
2640 TXT5       LDA #7
2645            STA FLAG2
2650 TXT6       LDX FLAG2
2655            LDA $200,X
2660            LDX FLAG1
2665            AND $BIT,X
2670            BEQ TXT8
2675            LDA X1
2680            LDY X1+1
2685            STA XVALUE
2690            STY XVALUE+1
2695            LDY Y1+1
2700            LDA FLAG2
2705            CLC
2710            ADC Y1
2715            STA $14
2720            BCC TXT7
2725            INY
2730 TXT7       JSR WORKDOT
2735 TXT8       DEC FLAG2
2740            BPL TXT6
2745            INC X1
2750            BNE TXT9
2755            INC X1+1
2760 TXT9       INC FLAG1
2765            LDX FLAG1
2770            CPX #8
2775            BNE TXT5
2780            INC FLAG3
2785            LDA FLAG3
2790            CMP LENGH
2795            BNE TXT10
2800            RTS
2805 TXT10      JMP TXT2
2810 STRING     JSR $AEFD
2815            JSR $AD9E
2820            JSR $B6A3
2825            LDY #0
2830            LDA ($64),Y
2835            STA $63
2840            INY
2845            LDA ($64),Y
2850            TAX
2855            INY
2860            LDA ($64),Y
2865            STA $65
2870            STX $64
2875            LDY #0
2880            LDA $63
2885            RTS
2890 SCANN      BIT LINETYP
2895            BMI SCA1
2900            RTS
2905 SCA1       LDX #$F0
2910            STX FLAG
2915            JSR WORKDOT
2920            LDA Y1
2925            LDY Y1+1
2930            STA $14
2935            BIT FLAG
2940            BPL SCA2
2945            PLA
2950            PLA
2955            RTS
2960 SCA2       LDX #0
2965            STX FLAG
2970            RTS
2975 SCANNLN    LDA #$FF
2980            JMP LINE+7
2985            ;**************************
2990            ;
2995            ; D A T E N B E R E I C H
3000            ;
3005            ;**************************
3010 BIT        .BY 128 64 32 16 8 4 2 1
3015 TABLE1     .BY 13 8 'A' 27 'B' 27
3020 TABLE2     .BY 10 13 '          ' 27 '*' 1 XMAX
3025 PROMPT     .BY 'EPSON-SUPPORT  V1.7  ***' 13
3030            .BY '(C)1984 BY ALEX DEL PINO' 13 0
3035 STMTAB     .BY 'CLEAR' 0
3040            .BY 'DOT' 0
3045            .BY 'RESETDOT' 0
3050            .BY 'INVDOT' 0
3055            .BY 'LINE' 0
3060            .BY 'RESETLINE' 0
3065            .BY 'INVLINE' 0
3070            .BY 153 0
3075            .BY 'EXIT' 0
3080            .BY 'TEXT' 0
3085            .BY 'INVTEXT' 0
3090            .BY 'RESETTEXT' 0
3095            .BY 'SCANLINE' 0 0
3100 ENTRY      .BY CLEAR-1 DOT-1 RESETDOT-1 INVERTDOT-1
3105            .BY LINE-1 RESETLINE-1 INVERTLINE-1
                    HARDCOPY-1 EXIT-1
3110            .BY TEXT-1 INVTEXT-1 RESTEXT-1 SCANNLN-1
3115            .EN ;END OF SOURCE-FILE

READY.
```

Listing 4. Der Source-Code zum Epson-Support (Schluß)

Plotter-Basic

Plotter-Basic ist eine Basic-Befehlserweiterung für den Printer/Plotter 1520 von Commodore. Es ermöglicht eine starke Vereinfachung der normalen Plotterbedienungen und erlaubt zusätzlich zum Beispiel räumliche Zeichnungen zu erstellen und beliebig zu drehen.

Außerdem verfügt es über die Möglichkeit Kreise beziehungsweise Ellipsen zu zeichnen, Blockdiagramme zu erstellen oder komfortable Funktionen-Plot-Programme zu schreiben.

Die Befehlserweiterung funktioniert nach dem Prinzip, daß jedes Befehlswort auf einen vorangestellten Punkt untersucht wird, der einen Plotter-Basic-Befehl kennzeichnet, zum Beispiel .RESET.

Nach dem Start mit SYS 12 * 4096 initialisiert sich das Programm (Listing 1) und gibt eine Einschaltmeldung aus. Die Initialisierung besteht darin, daß auf eventuelle andere Spracherweiterungen Rücksicht genommen wird und die Zeiger so verändert werden, daß die andere Erweiterung funktionstüchtig bleibt. Voraussetzung dafür ist natürlich, daß der Speicherbereich, den Plotter-Basic benötigt, nicht von fremden Programmen gebraucht wird. Es handelt sich hierbei um den Bereich $C000 bis $C8F5 beziehungsweise 49187 bis 51445.

Die Zeiger, die verändert werden mußten, sind die Vektoren $308/$309 und $300/$301 (Warmstart-Vektor). Normalerweise zeigt der Vektor $308/$309 auf die Routine $A7E4 (Basic-Befehlsadresse holen). Nach der Initialisierung zeigt er nun auf eine Routine, die auf einen vorangestellten Punkt testet und den gegebenenfalls erkannten Befehl ausführt.

Die Veränderung des Warmstartvektors ermöglichte es bei dem Befehl .PROG, der das Programmlisting zum Drucker schickt, die READY-Meldung wieder auf dem Bildschirm auszugeben und den Druckkanal ordnungsgemäß zu schließen. Außerdem besteht durch Veränderung dieses Zeigers die Möglichkeit, Fehlermeldungen zu unterdrücken beziehungsweise zu kontrollieren, ohne den Programmablauf abbrechen zu lassen.

Befehlsbeschreibung:

Die allgemeine Syntax der Befehle sieht folgendermaßen aus:

a) .BEFEHLSWORT keine Parameter erforderlich
oder
b) .BEFEHLSWORT,Parameter1,Parameter2,...,Parameter n

Sieht der Befehl also eine Parameterübergabe vor, so müssen diese vom Befehlswort durch Kommata getrennt werden. Die Parameter können sowohl Variablen als auch Zahlen sein. Ausnahmen sind bei den entsprechenden Beschreibungen erklärt.

Grundsätzlich muß jedem Plotter-Basic-Befehlswort ein Punkt vorangestellt werden. Mehrere Befehle in einer Zeile müssen durch einen Doppelpunkt voneinander getrennt sein.

Beispiel: .BLAU:.SUPER

Eine Besonderheit sei hier noch erwähnt, die ihre Ursache darin hat, daß die neuen Befehlsworte nicht als Tokens codiert wurden. Die neuen Befehle werden im Speicher nämlich Buchstabe für Buchstabe abgelegt. Basic-Befehle dagegen besitzen jeweils einen Ein-Byte-Code (Token), in den sie nach der Eingabe umgewandelt werden. Bevor aber nun der Basic-Interpreter die einzelnen Tokens erkennt, wird eine Untersuchung auf Plotter-Basic-Befehle durchgeführt. Diese Methode hat ihre Schwachstelle bei einer IF...THEN-Abfrage. Sobald nämlich der Befehl hinter THEN abgearbeitet werden soll, wird nicht mehr zur umgelenkten Interpreterschleife zurückgesprungen, sondern sofort der nächste Befehl als Token interpretiert. Falls dieser Befehl aber aus Plotter-Basic stammt, kommt es zu einem Syntax-Error. Dieses Problem läßt sich jedoch lösen, indem man dem THEN einen Doppelpunkt folgen läßt.

Beispiel: IF A=99 THEN:.TEXT,"neunundneunzig"

Möchte man Plotter-Basic mit einer anderen Spracherweiterung gleichzeitig betreiben, so muß Plotter-Basic als letztes geladen werden. Da unter Umständen manche Befehlsworte mit der anderen Sprache identisch sind, kann es zu Fehlermeldungen kommen, die man dadurch unterdrückt, daß man nur die ersten Buchstaben des Plotter-Basic-Befehls eingibt. Es wird dann der Befehl ausgeführt, der die entsprechenden Anfangsbuchstaben besitzt und als erstes im Speicher steht. Bei den meisten Befehlen reicht es, nur die beiden ersten Buchstaben zur Identifizierung anzugeben.

Beispiel: .RO = .ROT
.PR,10 = .PROG,10

Die Parameter können also auch schon nach zwei Buchstaben folgen. Die Reihenfolge, in der die Befehlsworte im Speicher abgelegt sind, und ihre Startadressen sind am Ende der Beschreibung aufgeführt. Dadurch ist es möglich, die Befehle über SYS-Aufrufe ausführen zu lassen.

Nun aber zur Erklärung der Befehle im Einzelnen:

1. Farbwahl

.BLAU
.ROT
.GRÜN
.SCHWARZ
} stellt die entsprechende Farbe ein

.FARBE,f wählt die Zeichenfarbe gemäß f.

Der Parameter f entspricht der Farbnummer, wie es in der Plotter-Anleitung erklärt ist.

2. Wahl der Zeichengröße

.KLEIN entspricht 80 Zeichen/Zeile
.MITTEL entspricht 40 Zeichen/Zeile
.GROSS entspricht 20 Zeichen/Zeile
.SUPER entspricht 10 Zeichen/Zeile
.SIZE,g wählt die Zeichengröße gemäß g.
(Näheres siehe Plotter-Anleitung.)

3. Ausdruck von ASCII-Daten

.TEXT,a$ druckt den String a$
.TEXT erzeugt eine Leerzeile (Carriage Return)
.PROG
.PROG,10
.PROG,40—
.PROG,100—556

Dieser Befehl entspricht dem LIST-Befehl in Basic. Ohne Parameter wird das gesamte Programm gelistet, mit Parametern wird entsprechend eine Zeile oder mehrere Zeilen ausgedruckt.

4. Wahl der Schriftarten

.GG stellt den Groß/Grafik-Modus ein.
.GK stellt den Groß/Klein-Modus ein.

5. Zeichendrehung

.DREH,r stellt für r=0 den Horizontalbetrieb ein und verursacht bei r=1 eine Zeichendrehung um 90 Grad nach rechts.

6. Druckerrückstellung und Speicherlöschung

.RESET setzt den Printer/Plotter zurück und löscht seinen Speicher.

7. Wahl der Linienart

.STRICH,l bestimmt die Linienart. Für l gelten die Angaben der Plotteranleitung.

8. Zeichnen von X/Y-Daten

.HOME bewegt zum Startpunkt (0/0).

.INIT legt den relativen Nullpunkt auf die derzeitige X/Y-Position.

.MOVE,x,y bewegt zur Position (x/y) relativ zum absoluten Nullpunkt (0/0).

.DRAW,x,y zeichnet zur Position (x/y) relativ zum absoluten Nullpunkt (0/0).

.RMOVE,x,y bewegt zur Position (x/y) relativ zum relativen Nullpunkt (x0/y0).

.RDRAW,x,y zeichnet zur Position (x/y) relativ zum relativen Nullpunkt (x0/y0).

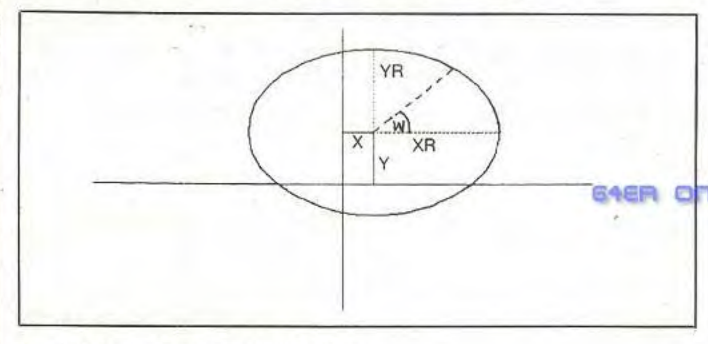
.LINIE,x1,y1,x2,y2 zeichnet eine Linie von Position (x1/y1) zu Position (x2/y2).

.RAHMEN,x,y,b,h zeichnet einen Rahmen ab dem Eckpunkt (x/y) mit der Breite b und der Höhe h.

.KREIS,x,y,xr,yr zeichnet eine Ellipse um den Mittelpunkt (x/y) mit einem Radius in x-Richtung von xr und einem Radius in y-Richtung von yr. Gilt xr=yr so entsteht ein Kreis.

.RADIUS,x,y,xr,yr,w zeichnet in die entsprechende Ellipse einen Radius, der mit der positiven X-Achse den Winkel w (in Grad!) bildet.

Erklärung zum KREIS- und RADIUS-Befehl:



9. Zeichnen von X/Y/Z-Daten

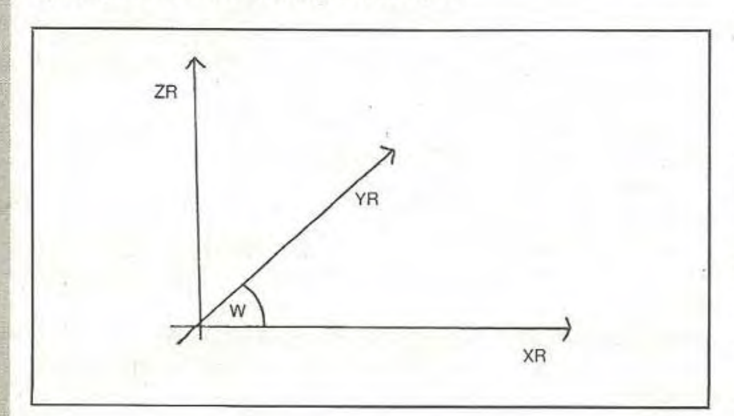
Die folgenden Befehle dienen dazu, dreidimensionale Grafiken zu erzeugen. Im wesentlichen transformieren sie drei Raumkoordinaten auf zwei Ebenenkoordinaten. Allen Berechnungen liegen diese Formeln zugrunde:

x=f*[xr+yr*cos(w)]+xv
y=f*[zr+yr*sin(w)]+yv

xr, yr und zr sind Raumkoordinaten, w ist der Betrachtungswinkel des Punktes in Grad, und f stellt einen Vergrößerungs- (für f größer als 1) beziehungsweise Verkleinerungsfaktor (für f zwischen 0 und 1) dar.

xv und yv dienen dazu, den Punkt im zweidimensionalen Koordinatensystem zu verschieben.

Die Achsen des räumlichen Koordinatensystems sind entsprechend der Abbildung bezeichnet:



.TRANSM,xr,yr,zr,xv,yv,f,w bewegt zur Position (x/y) relativ zum absoluten Nullpunkt, die sich durch Einsetzen aller Parameter in die obige Formel ergibt.

.TRANSD,xr,yr,zr,xv,yv,f,w zeichnet zur entsprechenden Position (x/y) relativ zum absoluten Nullpunkt.

.TRANSL,xr1,yr1,zr1,xv,yv,f,w,xr2,yr2,zr2 zieht eine Linie von (x1/y1) nach (x2/y2) entsprechend den Parametern.

Möchte man also beispielsweise das räumliche Koordinatensystem zeichnen, so wählt man als Ursprung der drei Achsen den Nullpunkt, also xr1=yr1=zr1=0. Je nachdem, welche Achse gezeichnet werden soll, setzt man für die xr-Achse xr2=1, yr2=0, zr2=0, für die yr-Achse dann xr2=0, yr2=1, zr2=0 und für die zr-Achse xr2=0, yr2=0 und zr2=1. Um das Bild in die Mitte des Papiers zu bringen, setzt man die x-Verschiebung xv=240 und die y-Verschiebung yv=0. Da als Länge für die Achsen »eins« zu klein ist, könnte man den Faktor f=50 wählen. Der Betrachtungswinkel sei w=45 (Grad). Somit ergibt sich für die Zeichnung des Koordinatensystems:

```
10 .TRANSL,0,0,0,240,0,50,45,1,0,0:REM xr-Achse
20 .TRANSL,0,0,0,240,0,50,45,0,1,0:REM yr-Achse
30 .TRANSL,0,0,0,240,0,50,45,0,0,1:REM zr-Achse
```

10. Drehung von x/y/z-Koordinaten

Um einen Körper von allen Seiten darstellen zu können, müssen die Raumkoordinaten gedreht werden. Dabei ist es natürlich möglich, um jede beliebige Raumachse und um jeden beliebigen Winkel zu drehen. Macht man sich eine solche Drehung am Koordinatensystem deutlich, so bemerkt man, daß sich jeweils die Komponente nicht verändert, um deren Achse gedreht werden soll. Bei einer Drehung um die xr-Achse verändern sich nur die yr- und die zr-Komponente, bei einer Drehung um die yr-Achse verändern sich nur xr und zr etc.

Der Befehl .TURN,k1,k2,w benötigt daher außer der Angabe des Drehwinkels w in Grad nur die zwei Komponenten k1 und k2. Hierbei muß es sich um Variablen handeln! Vor dem Befehlsaufruf befinden sich in den Variablen die räumlichen Koordinaten eines Punktes und nach der Befehlsausführung stehen in den Variablen die Koordinaten des neuen (gedrehten) Punktes.

Angenommen, in den Variablen xr,yr und zr befinden sich die Koordinaten eines Punktes, so könnte man ihn folgendermaßen um den Winkel w drehen:

Drehung um die xr-Achse: .TURN,yr,zr,w
Drehung um die yr-Achse: .TURN,xr,zr,w
Drehung um die zr-Achse: .TURN,xr,yr,w

Anhand der Formeln zur Drehung läßt sich erkennen, warum der Befehl für alle drei Drehrichtungen richtige Ergebnisse liefert:

Drehung um die xr-Achse:
xr'=xr yr'=yr*cos(w)+zr*sin(w)
zr'=—yr*sin(w)+zr*cos(w)

Drehung um die yr-Achse:
yr'=yr xr'=xr*cos(w)+zr*sin(w)
zr'=—xr*sin(w)+zr*cos(w)

Drehung um die zr-Achse:
zr'=zr xr'=xr*cos(w)+yr*sin(w)
yr'=—xr*sin(w)+yr*cos(w)

Ein kleines Beispielprogramm (Listing 2) demonstriert die Möglichkeiten dieses und des TRANSL-Befehls anhand der Drehung eines Körpers.

11. Fehlerbehandlung

Um zum Beispiel ein Programm zu schreiben, das jede beliebige Funktion plottet, ist es erforderlich, mathematische Fehlermeldungen zu unterdrücken. Sobald nämlich eine Funktion Definitionslücken besitzt, steigen die meisten Programme mit einem »division by zero« oder ähnlichem aus.

Der Befehl .ERRJUMP,ze setzt die Programmausführung in der Zeile ze fort, sobald ein arithmetischer Fehler aufgetreten

ist. Es handelt sich hierbei um die Fehler division by zero, illegal quantity und overflow. Sollte innerhalb einer FOR-NEXT-Schleife oder in einem Unterprogramm ein Fehler auftreten, so kann nicht wieder zu NEXT beziehungsweise RETURN gesprungen werden, da nun die notwendige Rücksprungadresse nicht mehr gefunden wird. FOR-NEXT-Schleifen und Unterprogrammsprünge lassen sich in solchen Fällen jedoch auch mit GOTO programmieren.

Der Befehl ist nur während des Programmablaufs wirksam, um zu verhindern, daß bei Fehlern im Direkt-Modus das Programm neu gestartet wird.

Sollen grundsätzlich alle Fehler unterdrückt werden, so muß hinter der Zeilennummer noch eine Sekundärvariable folgen:

.ERRJUMP,ze,s setzt die Programmausführung in der Zeile ze fort, sobald irgendein Fehler auftritt. Die Variable s ist nur eine Scheinvariable, das heißt sie wird weder verändert noch spielt ihr Typ eine Rolle. Am einfachsten ist es hier, eine Zahl anzugeben.

.ERRNUM,x schreibt in die Variable x die Fehlernummer des zuletzt aufgetretenen Fehlers. Bei x muß es sich um eine Variable handeln! Die Fehlernummer ist nur einmal lesbar und wird danach wieder auf Null gesetzt, bis der nächste Fehler auftritt. Innerhalb der Variablen bleibt die Fehlernummer natürlich erhalten. Die Fehlermeldungen besitzen folgende Nummern:

1	too many files	2	file open
3	file not open	4	file not found
5	device not present	6	not input file
7	missing filename	8	missing filename
9	illegal dev. number	10	next without for
11	syntax	12	return without gosub
13	out of data	14	illegal quantity
15	overflow	16	out of memory
17	undef'd statement	18	bad subscript
19	redim'd array	20	division by zero
21	illegal direct error	22	type mismatch
23	string too long	24	file data
25	form. too complex	26	can't continue
27	undef'd function	28	verify
29	load		

.ERROUT schaltet innerhalb des Programms die Fehlerunterdrückung wieder aus. Dieser Befehl wird zum Beispiel dann nötig, wenn am Ende des Programms ein Listing ausgedruckt werden soll, da beide Befehle den Warmstartvektor ändern. Grundsätzlich wird die Fehlerunterdrückung mit Ende des Programms abgeschaltet.

12. Ausschalten von Plotter-Basic

.AUS setzt alle Zeiger wieder so, wie sie vor dem Start von Plotter-Basic standen und schaltet somit die Spracherweiterung ab. Erneutes Einschalten geschieht mit SYS 12 * 4096.

Die Befehlsadressen im Speicher

Um sämtliche Zeiger unverändert zu lassen, kann man Plotter-Basic auch ohne Startaufruf benutzen. Nach dem Laden lassen sich alle Befehle über SYS-Aufrufe ausführen. Zum Beispiel: .KREIS,240,0,80,50 entspricht SYS50184, 240,0,80,50

Parameter werden wie gewöhnlich durch das Komma getrennt.

Die Reihenfolge entspricht der Reihenfolge, in der die Befehlsworte erkannt werden. .KREIS wird also schneller erkannt als .FARBE.

INIT	= 49916	RADIUS	= 50644
SIZE	= 49783	ERROUT	= 51290
MOVE	= 49926	TRANSM	= 50729
DREH	= 49833	ERRNUM	= 51379
DRAW	= 49974	TRANSD	= 50751
STRICH	= 49888	AUS	= 49157
RMOVE	= 49987	TRANSL	= 50764
GG	= 49807	TURN	= 51091
RDRAW	= 50000	BLAU	= 49695
GK	= 49815	ROT	= 49703
HOME	= 49906	GRUEN	= 49711
PROG	= 50413	SCHWARZ	= 49719
RAHMEN	= 50471	FARBE	= 49727
TEXT	= 49851	KLEIN	= 49751
LINIE	= 50013	MITTEL	= 49759
RESET	= 49823	GROSS	= 49767
KREIS	= 50184	SUPER	= 49775
ERRJUMP	= 51233		

(Stephan Schäfer/gk)

```
10 REM ******************                    <053>
20 REM PLOTTER-BASIC DEMO                    <093>
30 REM ******************                    <073>
35 .SCHWARZ:.GK:.MITTEL                      <183>
37 .TEXT,"{5SPACE}DREHUNG EINES 3-D-KOERPE
   RS"                                       <120>
38 .TEXT,"{5SPACE}======================
   =="                                       <156>
39 .MOVE,0,-70:.TEXT                         <162>
40 XV=240:YV=0:REM VERSCHIEBUNGSVEKTOR       <022>
41 F=15        :REM VERGROESSERUNG           <236>
42 W=35        :REM BETRACHTUNGSWINKEL       <012>
43 L=24        :REM ANZAHL DER LINIEN        <079>
45 DIM R(L)    :REM FELD DER LINIEN          <161>
50 P=8         :REM ANZAHL DER ECKPUNKTE     <061>
55 DIM X(P):DIM Y(P):DIM Z(P)                <115>
60 REM EINLESEN DER ECKPUNKTE                <191>
70 FOR I=1 TO P                              <056>
80 READ X(I),Y(I),Z(I)                       <054>
90 NEXT                                      <100>
100 REM EINLESEN DER VERBINDUNGSFOLGE        <022>
110 FOR I=1 TO L                             <094>
120 READ R(I)                                <029>
130 NEXT                                     <140>
140 REM ZEICHNUNG                            <120>
150 FOR I=1 TO L STEP 2                      <147>
160 .TRANSL,X(R(I)),Y(R(I)),Z(R(I)),XV,YV,
    F,W,X(R(I+1)),Y(R(I+1)),Z(R(I+1))        <011>
170 NEXT                                     <180>
180 REM DREHUNG UM 10 GRAD                   <020>
185 DW=DW+10                                 <242>
190 FOR I=1 TO P                             <176>
200 .TURN,X(I),Y(I),DW:REM DREHUNG UM Z      <111>
210 NEXT                                     <220>
220 .MOVE,0,-150                             <059>
230 .TEXT             :REM CR SENDEN         <231>
240 PRINT "{DOWN}COMMODORE-TASTE DRUECKEN!
    "                                        <174>
250 WAIT 653,2                               <210>
260 PRINT "{UP,25SPACE,UP}"                  <080>
270 GOTO 140                                 <008>
499 REM ECKPUNKTE                            <023>
500 DATA -2,-2,0,2,-2,0,2,2,0,-2,2,0,0,0,6
    ,4,0,6,4,4,6,0,4,6                       <155>
599 REM REIHENFOLGE DER VERBINDUNGEN         <103>
600 DATA 1,2,2,3,3,4,4,1,1,5,2,6,3,7,4,8,8
    ,5,5,6,6,7,7,8                           <101>
```

© 64'er

Listing 2. Dieses Programm dient dazu, einen Spat in 10-Grad-Schritten zu drehen und dann zu zeichnen. Das neue Bild wird jeweils nach Drücken der Commodore-Taste gekennzeichnet (Programm-Abbruch über RUN/STOP-RESTORE).

```
programm : Pl-Basic      c000 c8d7
-----------------------------------
c000 : ea 20 64 c3 60 ad fa c3   73
c008 : 8d 00 03 ad fb c3 8d 01   22
c010 : 03 ad 38 c0 8d 08 03 ad   90
c018 : 39 c0 8d 09 03 a9 ea 8d   7a
c020 : 00 c0 60 20 73 00 c9 2e   57
c028 : f0 10 38 a5 7a e9 01 85   e9
c030 : 7a a5 7b e9 00 85 7b 4c   4c
```

Listing 1. Plotter-Basic muß mit dem MSE (auf Seite 8) eingegeben werden

```
c038 : 00 00 a9 d7 85 fb a9 c8   0e
c040 : 85 fc a9 b8 85 fd a9 c0   35
c048 : 85 fe 20 73 00 20 56 c0   9f
c050 : 20 70 c0 4c ae a7 a0 00   0d
c058 : 91 fb c8 20 73 00 f0 0b   2e
c060 : c9 2c f0 07 c9 3a f0 03   95
c068 : 4c 58 c0 a9 00 91 fb 60   83
c070 : a2 00 a0 00 b1 fb f0 08   09
c078 : d1 fd d0 07 c8 4c 74 c0   9f
c080 : 4c 9d c0 e8 e0 23 f0 12   f7
c088 : 18 a9 07 65 fd 85 fd a9   3b
c090 : 00 65 fe 85 fe a0 00 4c   41
c098 : 74 c0 4c 08 af 8a 18 0a   44
c0a0 : 69 ad 85 fb a9 c1 69 00   0f
c0a8 : 85 fc a0 00 b1 fb 85 fd   e0
c0b0 : c8 b1 fb 85 fe 6c fd 00   4c
c0b8 : 49 4e 49 54 20 20 20 4d   23
c0c0 : 4f 56 45 20 20 20 44 52   48
c0c8 : 41 57 20 20 20 52 4d 4f   29
c0d0 : 56 45 20 20 52 44 52 41   e8
c0d8 : 57 20 20 48 4f 4d 45 20   05
c0e0 : 20 20 52 41 48 4d 45 4e   6e
c0e8 : 20 4c 49 4e 49 45 20 20   ca
c0f0 : 4b 52 45 49 53 20 20 52   3a
c0f8 : 41 44 49 55 53 20 54 52   85
c100 : 41 4e 53 4d 20 54 52 41   57
c108 : 4e 53 44 20 54 52 41 4e   8e
c110 : 53 4c 20 54 55 52 4e 20   7d
c118 : 20 20 42 4c 41 55 20 20   e2
c120 : 20 52 4f 54 20 20 20 20   8c
c128 : 47 52 55 45 4e 20 20 53   a3
c130 : 43 48 57 41 52 5a 46 41   29
c138 : 52 42 45 20 20 4b 4c 45   19
c140 : 49 4e 20 20 4d 49 54 54   d6
c148 : 45 4c 20 47 52 4f 53 53   38
c150 : 20 20 53 55 50 45 52 20   b9
c158 : 20 53 49 5a 45 20 20 20   d6
c160 : 44 52 45 48 20 20 20 53   52
c168 : 54 52 49 43 48 20 47 47   d1
c170 : 20 20 20 20 20 47 4b 20   56
c178 : 20 20 20 20 50 52 4f 47   18
c180 : 20 20 20 54 45 58 54 20   ec
c188 : 20 20 52 45 53 45 54 20   e7
c190 : 20 45 52 52 4a 55 4d 50   57
c198 : 45 52 52 4f 55 54 20 45   88
c1a0 : 52 52 4e 55 4d 20 41 55   df
c1a8 : 53 20 20 20 20 fc c2 06   18
c1b0 : c3 36 c3 43 c3 50 c3 f2   9b
c1b8 : c2 27 c5 5d c3 08 c4 d4   64
c1c0 : c5 29 c6 3f c6 4c c6 93   c4
c1c8 : c7 1f c2 27 c2 2f c2 37   d3
c1d0 : c2 3f c2 57 c2 5f c2 67   ce
c1d8 : c2 6f c2 77 c2 a9 c2 e0   38
c1e0 : c2 8f c2 97 c2 ed c4 bb   33
c1e8 : c2 9f c2 21 c8 5a c8 b3   38
c1f0 : c8 05 c0 a0 02 20 01 c2   2a
c1f8 : a5 ff 20 d2 ff 20 16 c2   de
c200 : 60 a2 06 a9 63 20 ba ff   8a
c208 : a9 00 20 bd ff 20 c0 ff   75
c210 : a2 63 20 c9 ff 60 20 cc   c2
c218 : ff a9 63 20 c3 ff 60 a9   da
c220 : 31 85 ff 20 f3 c1 60 a9   3a
c228 : 33 85 ff 20 f3 c1 60 a9   44
```

```
c230 : 32 85 ff 20 f3 c1 60 a9   4b
c238 : 30 85 ff 20 f3 c1 60 20   3e
c240 : fd ae a0 02 20 01 c2 20   52
c248 : 9e ad 20 fc c3 20 16 c2   80
c250 : 60 a0 03 20 f5 c1 60 a9   07
c258 : 30 85 ff 20 51 c2 60 a9   4f
c260 : 31 85 ff 20 51 c2 60 a9   58
c268 : 32 85 ff 20 51 c2 60 a9   61
c270 : 33 85 ff 20 51 c2 60 20   57
c278 : fd ae a0 03 20 01 c2 20   aa
c280 : 9e ad 20 fc c3 20 16 c2   b8
c288 : 60 a0 06 20 f5 c1 60 a9   00
c290 : 30 85 ff 20 89 c2 60 a9   0a
c298 : 31 85 ff 20 89 c2 60 a0   01
c2a0 : 07 a9 00 85 ff 20 f5 c1   89
c2a8 : 60 20 fd ae 20 9e ad a0   5c
c2b0 : 04 20 01 c2 20 fc c3 20   96
c2b8 : 16 c2 60 a9 2c a0 00 d1   e8
c2c0 : 7a d0 0f 20 73 00 a0 00   24
c2c8 : 20 01 c2 20 a4 aa 20 16   6a
c2d0 : c2 60 a0 00 20 01 c2 a9   53
c2d8 : 0d 20 d2 ff 20 16 c2 60   28
c2e0 : 20 fd ae a0 05 20 01 c2   99
c2e8 : 20 9e ad 20 fc c3 20 16   61
c2f0 : c2 60 a0 01 a9 48 85 ff   1e
c2f8 : 20 f5 c1 60 a0 01 a9 49   da
c300 : 85 ff 20 f5 c1 60 20 fd   e7
c308 : ae a0 01 20 01 c2 a9 4d   b2
c310 : 20 d2 ff a9 20 20 d2 ff   1d
c318 : 20 9e ad 20 dd bd 20 1e   7f
c320 : ab a9 20 20 d2 ff 20 fd   56
c328 : ae 20 9e ad 20 dd bd 20   6c
c330 : 1e ab 20 16 c2 60 20 fd   9a
c338 : ae a0 01 20 01 c2 a9 44   d0
c340 : 4c 10 c3 20 fd ae a0 01   63
c348 : 20 01 c2 a9 52 4c 10 c3   1e
c350 : 20 fd ae a0 01 20 01 c2   c9
c358 : a9 4a 4c 10 c3 20 06 c3   18
c360 : 20 36 c3 60 a2 ff e8 bd   e1
c368 : a8 c3 20 d2 ff e0 51 d0   42
c370 : f5 ad 00 03 8d fa c3 ad   b7
c378 : 01 03 8d fb c3 18 a9 03   87
c380 : ad 08 03 8d 38 c0 a9 00   d4
c388 : ad 09 03 8d 39 c0 a9 0d   87
c390 : 8d 00 03 a9 c5 8d 01 03   e6
c398 : a9 23 8d 08 03 a9 c0 8d   d3
c3a0 : 09 03 a9 60 8d 00 c0 60   3e
c3a8 : 93 11 05 20 20 20 2a 2a   09
c3b0 : 2a 2a 20 43 4f 4d 4d 4f   93
c3b8 : 44 4f 52 45 20 31 35 32   a6
c3c0 : 30 20 42 41 53 49 43 20   86
c3c8 : 31 2e 33 20 2a 2a 2a 2a   d2
c3d0 : 0d 0d 20 20 20 28 50 29   47
c3d8 : 20 4a 41 4e 2e 31 39 38   f9
c3e0 : 35 20 3d 3a 3d 20 28 43   b8
c3e8 : 29 20 42 59 20 53 2e 53   d9
c3f0 : 43 48 41 45 46 45 52 20   68
c3f8 : 20 0d 00 00 20 dd bd 20   c7
c400 : 1e ab a9 20 20 d2 ff 60   bc
c408 : 20 12 c4 20 47 c4 20 6c   5a
c410 : c4 60 20 fd ae 20 9e ad   8e
c418 : a2 d7 a0 c8 20 d4 bb 20   bf
```

Listing 1. Plotter-Basic (Fortsetzung)

```
c420 : fd ae 20 9e ad a2 dc a0   f5
c428 : c8 20 d4 bb 20 fd ae 20   9a
c430 : 9e ad a2 e1 a0 c8 20 d4   04
c438 : bb 20 fd ae 20 9e ad a2   4b
c440 : e6 a0 c8 20 d4 bb 60 a9   ac
c448 : d7 a0 c8 20 a2 bb a0 01   32
c450 : 20 01 c2 a9 4d 20 d2 ff   f8
c458 : 20 fc c3 a9 dc a0 c8 20   53
c460 : a2 bb 20 fc c3 20 16 c2   a3
c468 : 20 fc c2 60 a9 e1 a0 c8   81
c470 : 20 a2 bb a0 01 20 01 c2   7f
c478 : a9 52 20 d2 ff 20 fc c3   29
c480 : a9 30 20 d2 ff 20 16 c2   83
c488 : a9 00 8d f0 c8 8d f1 c8   05
c490 : 8d f2 c8 8d f3 c8 8d f4   20
c498 : c8 4c ca c4 a0 01 20 01   66
c4a0 : c2 20 64 e2 a9 e1 a0 c8   a5
c4a8 : 20 28 ba a9 4a 20 d2 ff   b1
c4b0 : 20 fc c3 a9 f0 a0 c8 20   ec
c4b8 : a2 bb 20 6b e2 a9 e6 a0   06
c4c0 : c8 20 28 ba 20 fc c3 20   33
c4c8 : 16 c2 a9 6c a0 e3 20 a2   26
c4d0 : bb a9 f0 a0 c8 20 67 b8   4d
c4d8 : a2 f0 a0 c8 20 d4 bb a9   1e
c4e0 : c9 a0 c5 18 20 5b bc b0   9f
c4e8 : b3 20 16 c2 60 a9 2c a0   ce
c4f0 : 00 d1 7a d0 03 20 73 00   90
c4f8 : a0 00 20 01 c2 c6 7a a9   60
c500 : 9b a2 63 8e ce c5 20 f7   22
c508 : a7 20 16 c2 60 ad ce c5   d7
c510 : c9 63 f0 03 6c fa c3 a9   28
c518 : 00 8d ce c5 a9 0d 20 d2   74
c520 : ff 20 16 c2 4c 74 a4 20   48
c528 : 12 c4 a0 01 20 01 c2 a9   4d
c530 : 4d 20 d2 ff a9 20 20 d2   04
c538 : ff a9 d7 a0 c8 20 a2 bb   a5
c540 : 20 fc c3 a9 dc a0 c8 20   3b
c548 : a2 bb 20 fc c3 20 16 c2   8b
c550 : 20 b9 c5 a9 e1 a0 c8 20   7a
c558 : a2 bb 20 fc c3 a9 dc a0   be
c560 : c8 20 a2 bb 20 fc c3 20   92
c568 : 16 c2 20 b9 c5 a9 e1 a0   91
c570 : c8 20 a2 bb 20 fc c3 a9   b5
c578 : e6 a0 c8 20 a2 bb 20 fc   67
c580 : c3 20 16 c2 20 b9 c5 a9   6b
c588 : d7 a0 c8 20 a2 bb 20 fc   68
c590 : c3 a9 e6 a0 c8 20 a2 bb   85
c598 : 20 fc c3 20 16 c2 20 b9   97
c5a0 : c5 a9 d7 a0 c8 20 a2 bb   d3
c5a8 : 20 fc c3 a9 dc a0 c8 20   a3
c5b0 : a2 bb 20 fc c3 20 16 c2   f3
c5b8 : 60 a0 01 20 01 c2 a9 44   02
c5c0 : 20 d2 ff a9 20 20 d2 ff   cd
c5c8 : 60 83 55 dc a7 6e 00 7b   c0
c5d0 : 0e fa 35 12 20 12 c4 20   d1
c5d8 : fd ae 20 9e ad a9 cf a0   b1
c5e0 : c5 20 28 ba a2 f0 a0 c8   dc
c5e8 : 20 d4 bb 20 47 c4 a0 01   84
c5f0 : 20 01 c2 a9 4a 20 d2 ff   68
c5f8 : a9 20 20 d2 ff a9 f0 a0   66
c600 : c8 20 a2 bb 20 64 e2 a9   fc
c608 : e1 a0 c8 20 28 ba 20 fc   42
c610 : c3 a9 f0 a0 c8 20 a2 bb   87
c618 : 20 6b e2 a9 e6 a0 c8 20   b2
c620 : 28 ba 20 fc c3 20 16 c2   68
c628 : 60 20 7c c6 a0 01 20 01   25
c630 : c2 a9 4d 20 d2 ff a9 20   32
c638 : 20 d2 ff 20 d8 c6 60 20   4b
c640 : 7c c6 a0 01 20 01 c2 a9   d0
c648 : 44 4c 33 c6 20 7c c6 20   99
c650 : 2c c6 20 fd ae 20 9e ad   69
c658 : a2 d7 a0 c8 20 d4 bb 20   ff
c660 : fd ae 20 9e ad a2 dc a0   35
c668 : c8 20 d4 bb 20 fd ae 20   da
c670 : 9e ad a2 e1 a0 c8 20 d4   44
c678 : bb 4c 42 c6 20 fd ae 20   b0
c680 : 9e ad a2 d7 a0 c8 20 d4   13
c688 : bb 20 fd ae 20 9e ad a2   9b
c690 : dc a0 c8 20 d4 bb 20 fd   9a
c698 : ae 20 9e ad a2 e1 a0 c8   01
c6a0 : 20 d4 bb 20 fd ae 20 9e   30
c6a8 : ad a2 e6 a0 c8 20 d4 bb   cc
c6b0 : 20 fd ae 20 9e ad a2 eb   38
c6b8 : a0 c8 20 d4 bb 20 fd ae   71
c6c0 : 20 9e ad a2 f0 a0 c8 20   66
c6c8 : d4 bb 20 fd ae 20 9e ad   03
c6d0 : a2 f5 a0 c8 20 d4 bb 60   06
c6d8 : a9 f5 a0 c8 20 a2 bb a9   16
c6e0 : cf a0 c5 20 28 ba 20 64   16
c6e8 : e2 a9 dc a0 c8 20 28 ba   8e
c6f0 : a9 d7 a0 c8 20 67 b8 a9   39
c6f8 : f0 a0 c8 20 28 ba a9 e6   3b
c700 : a0 c8 20 67 b8 20 fc c3   01
c708 : a9 f5 a0 c8 20 a2 bb a9   46
c710 : cf a0 c5 20 28 ba 20 6b   54
c718 : e2 a9 dc a0 c8 20 28 ba   be
c720 : a9 e1 a0 c8 20 67 b8 a9   6e
c728 : f0 a0 c8 20 28 ba a9 eb   75
c730 : a0 c8 20 67 b8 20 fc c3   31
c738 : 20 16 c2 60 20 fd ae 20   0d
c740 : 8b b0 8d e6 c8 8c e7 c8   86
c748 : 20 fd ae 20 8b b0 8d e8   5d
c750 : c8 8c e9 c8 20 fd ae 20   df
c758 : 9e ad a2 f5 a0 c8 20 d4   af
c760 : bb 60 a9 f5 a0 c8 20 a2   8a
c768 : bb a9 cf a0 c5 20 28 ba   73
c770 : 20 64 e2 a2 eb a0 c8 20   f6
c778 : d4 bb a9 f5 a0 c8 20 a2   69
c780 : bb a9 cf a0 c5 20 28 ba   8b
c788 : 20 6b e2 a2 f0 a0 c8 20   e2
c790 : d4 bb 60 20 3c c7 20 62   a5
c798 : c7 ad e6 c8 ac e7 c8 20   76
c7a0 : a2 bb a9 eb a0 c8 20 28   29
c7a8 : ba a2 e1 a0 c8 20 d4 bb   98
c7b0 : ad e8 c8 ac e9 c8 20 a2   44
c7b8 : bb a9 f0 a0 c8 20 28 ba   3c
c7c0 : a9 e1 a0 c8 20 67 b8 a2   00
c7c8 : fa a0 c8 20 d4 bb ad e6   f8
c7d0 : c8 ac e7 c8 20 a2 bb 20   47
c7d8 : b4 bf a9 f0 a0 c8 20 28   15
c7e0 : ba a2 e1 a0 c8 20 d4 bb   d0
c7e8 : ad e8 c8 ac e9 c8 20 a2   7c
c7f0 : bb a9 eb a0 c8 20 28 ba   32
c7f8 : a9 e1 a0 c8 20 67 b8 ad   4e
c800 : e8 c8 ac e9 c8 85 49 84   9b
```

Listing 1. Plotter-Basic (Fortsetzung)

```
c808 : 4a 20 d6 a9 a9 fa a0 c8  d4
c810 : 20 a2 bb ad e6 c8 ac e7  5d
c818 : c8 85 49 84 4a 20 d6 a9  da
c820 : 60 20 fd ae 20 8a ad 20  33
c828 : f7 b7 a5 14 a4 15 8d 58  c1
c830 : c8 8c 59 c8 a9 65 8d 00  aa
c838 : 03 a9 c8 8d 01 03 a0 00  9e
c840 : a9 2c d1 7a d0 0c 20 73  98
c848 : 00 20 9e ad a9 63 8d a6  ef
c850 : c8 60 a9 00 8d a6 c8 60  a5
c858 : 00 00 a9 0d a0 c5 8d 00  d3
c860 : 03 8c 01 03 60 c9 81 d0  46
c868 : 0a c0 a3 d0 06 20 5a c8  32
c870 : 4c 91 c8 ad a6 c8 d0 1c  99
c878 : 86 02 8a 30 14 e0 0e f0  0a
c880 : 13 e0 0f f0 0f e0 10 f0  ff
c888 : 1e e0 14 f0 07 aa 6c fa  a7
c890 : c3 4c 74 a4 86 02 ad 58  0b
c898 : c8 ac 59 c8 85 14 84 15  5b
c8a0 : 20 a3 a8 4c ae a7 00 68  3e
c8a8 : a8 68 a2 fa 9a 48 98 48  6b
c8b0 : 4c 94 c8 20 fd ae 20 8b  69
c8b8 : b0 8d e6 c8 8c e7 c8 a4  76
c8c0 : 02 20 a2 b3 ad e6 c8 ac  80
c8c8 : e7 c8 85 49 84 4a 20 d6  67
c8d0 : a9 a9 00 85 02 60 00 00  22
```

Listing 1. Plotter-Basic (Schluß)

29 Druckerbefehle für Epson

Oft ist es recht mühsam, die vielen Fähigkeiten der Epson-Drucker auszunutzen. »Epson 29« macht das ständige Nachschlagen im Handbuch überflüssig.

Diese kurze, aber doch recht wirkungsvolle Erweiterung ermöglicht es auf einfachste Weise, Steuercodes an einen Epson-Drucker zu senden. Es belegt den freien RAM-Bereich zwischen den Speicherzellen 50000 und 50503, und arbeitet auch mit dem Centronics-Interface, das Sie in dieser Ausgabe finden, zusammen. Doch hier zuerst eine alphabetisch geordnete Übersicht:

- **← A** — Adresse: ← A, gefolgt von einer Zahl, legt fest, über welchen Kanal die folgenden Kommandos zum Drucker gesendet werden sollen. Beispiel: ← A4, die nachfolgenden Befehle werden über Kanal 4 zum Drucker geschickt. Wird dieser Befehl nicht gesendet, geht der Computer davon aus, daß Sie Kanal 1 benutzen.
- **← B** — Breitschrift an
- **← D** — Doppeldruck an
- **← E** — Eliteschrift an
- **← F** — Fettdruck an
- **← G** — Glocke, sendet Bell-Code an Drucker, um zum Beispiel den Druckbeginn akustisch anzukündigen.
- **← I** — Indexschrift an
- **← K** — Kursivschrift an
- **← L** — Linken Rand setzen. Beispiel: ← L15 setzt linken Rand auf die 15. Druckposition.
- **← N** — unidirektionaler Ausdruck (verbessert Qualität des Schriftbildes).
- **← O** — Potenzieren an
- **← P** — Pica an
- **← R** — Rechten Rand setzen (parallel zu ← L)
- **← S** — Schmalschrift an
- **← U** — Unterstreichen an
- **← 6** — Zeilenabstand auf 1/6 Zoll
- **← 7** — Zeilenabstand auf 1/72 Zoll
- **← 8** — Zeilenabstand auf 1/8 Zoll
- **← 2** — Zeilenabstand auf n/216 Zoll. Beispiel: ←2 159 setzt Zeilenabstand auf 159/216 Zoll.
- **← @** — Drucker-Reset
- **← 0B** — Breitschrift aus
- **← 0D** — Doppeldruck aus
- **← 0F** — Fettdruck aus
- **← 0I** — Indizieren aus
- **← 0K** — Kursivschrift aus
- **← 0N** — unidirektionaler Druck aus
- **← 0O** — Potenzieren aus
- **← 0S** — Schmalschrift aus
- **← 0U** — Unterstreichmodus aus

Als Kennzeichen für einen neuen Befehl steht ein Pfeil nach links (Taste in der linken oberen Ecke der Tastatur). Anstelle dieses Pfeiles kann man jedoch auch ein anderes Zeichen verwenden (zum Beispiel »!«), indem man in die Speicherzelle 50020 den ASCII-Wert dieses Zeichens POKEt. Falls Sie einen nicht definierten Wert senden, dies wäre zum Beispiel —*, gibt der Computer die Fehlermeldung »PRINTER COMMAND ERROR« aus.

Hier noch zwei weitere Vorteile des Programms:

1. Die Befehle sind bis zu 6mal schneller. Um auf Breitschrift umzuschalten, benötigt der C 64 mittels CHR$-Codes zirka 0,013 Sekunden, mit dem Befehl ←B sind es nur noch 0,002 Sekunden.

2. Der Basic-Speicherplatzbedarf ist wesentlich geringer! So benötigt das Demoprogramm 1 (Listing 2) im Vergleich zu dem in Standard-Basic geschriebenen Demoprogramm 2 (Listing 3) 66 Bytes weniger, dies sind nur 57% des benötigten Speicherplatzes von Listing 2.

Ich empfehle Ihnen zum besseren Verständnis, sich einmal die Listings der Demoprogramme anzuschauen, um den Umgang mit den neuen Befehlen kennenzulernen.

Achtung:

Wenn einem Druckerbefehl kein PRINT#-Befehl folgt (eigentlich unlogisch, denn warum sollte man einen Befehl an den Drucker senden, wenn man gar nichts ausdrucken will?), befindet man sich im CMD-Modus. Angenommen, Sie geben ←F und danach LIST ein, erscheint das Listing auf dem Drucker statt auf dem Bildschirm.

Wollen Sie einen dieser neuen Befehle innerhalb einer IF...THEN-Schleife verwenden, so müssen Sie nach THEN einen Doppelpunkt setzen (Beispiel: IF A=1 THEN:←S)

Haben Sie den Basic-Lader eingetippt, können Sie das Programm als reinen Maschinencode auf Diskette oder Kassette abspeichern; geben Sie dazu folgendes im Direktmodus ein:

```
POKE 43,80
POKE 44,195
POKE 45,22
POKE 46,197
SAVE "DRUCKERBEF.MC",8,1 (für Diskette)
oder
SAVE "DRUCKERBEF.MC",1,1 (für Kassette)
```

(Olaf Amblank/gk)

```
programm : epson29          c350 c516
-----------------------------------

c350 : a9 01 8d 44 c5 a9 60 a0  d2
c358 : c3 8d 08 03 8c 09 03 60  22
c360 : 20 73 00 c9 5f f0 49 20  56
c368 : 79 00 4c e7 a7 a9 d1 8d  1b
c370 : 46 c5 4c eb c4 a9 6c 8d  90
c378 : 46 c5 4c eb c4 a0 d3 20  12
c380 : e0 c4 a9 00 4c da c4 a0  1d
c388 : d3 20 e0 c4 a9 01 4c da  c5
c390 : c4 a0 30 4c be c4 a0 31  31
c398 : 4c be c4 a0 32 4c be c4  93
c3a0 : a9 33 8d 46 c5 4c eb c4  07
c3a8 : 20 ca c4 a9 07 4c da c4  5b
c3b0 : 20 73 00 c9 32 f0 e9 c9  a9
c3b8 : 36 f0 e0 c9 37 f0 d7 c9  c5
c3c0 : 38 f0 ce c9 49 f0 c0 c9  10
c3c8 : 4c f0 aa c9 52 f0 9e c9  2b
c3d0 : 40 f0 70 c9 4f f0 a6 c9  88
c3d8 : 46 f0 6d c9 44 f0 6e c9  44
c3e0 : 45 f0 56 c9 50 f0 57 c9  ea
c3e8 : 53 f0 71 c9 55 f0 31 c9  7e
c3f0 : 4b f0 41 c9 4e f0 33 c9  0a
c3f8 : 42 f0 1b c9 41 f0 0b c9  0e
c400 : 30 f0 7d c9 47 f0 a1 4c  5c
c408 : fe c4 20 73 00 20 9e b7  ca
c410 : 8e 44 c5 4c ae a7 a0 d7  16
c418 : 20 e0 c4 a9 01 4c da c4  76
c420 : a0 2d 20 e0 c4 a9 01 4c  b1
c428 : da c4 a0 d5 20 e0 c4 a9  b7
c430 : 01 4c da c4 a0 34 4c be  01
c438 : c4 a0 cd 4c be c4 a0 d0  7f
c440 : 4c be c4 a0 40 4c be c4  1b
c448 : a0 c5 4c be c4 a0 c7 4c  bf
c450 : be c4 a0 c6 4c be c4 a0  80
c458 : c8 4c be c4 20 ca c4 a9  4d
c460 : 0f 4c da c4 20 ca c4 a9  a3
c468 : 12 4c da c4 a0 d4 4c be  4f
c470 : c4 a0 35 4c be c4 a0 d5  9b
c478 : 20 e0 c4 a9 00 4c da c4  c6
c480 : 20 73 00 c9 46 f0 cb c9  42
c488 : 4e f0 eb c9 44 f0 c8 c9  05
c490 : 53 f0 d1 c9 4b f0 da c9  44
c498 : 55 f0 19 c9 42 f0 0b c9  50
c4a0 : 4f f0 c9 c9 49 f0 c5 4c  df
c4a8 : fe c4 a0 d7 20 e0 c4 a9  9b
c4b0 : 00 4c da c4 a0 2d 20 e0  db
c4b8 : c4 a9 00 4c da c4 20 ca  c4
c4c0 : c4 20 d4 c4 98 a0 00 4c  89
c4c8 : da c4 ae 44 c5 20 c9 ff  bd
c4d0 : 20 73 00 60 a9 1b 20 d2  50
c4d8 : ff 60 20 d2 ff 4c ae a7  d6
c4e0 : 20 ca c4 20 d4 c4 98 20  b1
c4e8 : d2 ff 60 20 ca c4 20 9e  66
c4f0 : b7 20 d4 c4 ad 46 c5 20  e9
c4f8 : d2 ff 8a 4c da c4 a9 07  7e
c500 : 85 22 a9 c5 4c 45 a4 50  db
c508 : 52 49 4e 54 45 52 20 43  0b
c510 : 4f 4d 4d 41 4e c4 64 ff  1e
```

Listing 1. 29 Druckerbefehle. Bitte mit dem MSE auf Seite 8 eingeben.

```
1 REM **********************************      <132>
3 REM ** DEMO 2 - NORMALE CHR$-CODE ***       <155>
5 REM **********************************      <136>
10 :                                          <242>
100 OPEN 1,16:REM *DRUCKERKANAL OEFFNEN*      <131>
110 PRINT#1,CHR$(27);"E";:REM *FETTDRUCK A
    N*                                        <177>
120 PRINT#1,CHR$(27);"G";:REM *DOPPELDRUCK
     AN*                                      <064>
125 PRINT#1,CHR$(27);CHR$(108);CHR$(10);:R
    EM *LINKEN RAND AUF 10 SETZEN*            <130>
130 PRINT#1,"X";                              <213>
135 PRINT#1,CHR$(27);"S";CHR$(0);:REM *POT
    EIZIEREN AN*                              <234>
140 PRINT#1,"2";                              <085>
150 PRINT#1,CHR$(27);"T";:REM *POTENZIEREN
     AUS*                                     <037>
160 PRINT#1,"=256"                            <034>
```

© 64'er

Listing 3. Demo mit normalen Druckerbefehlen

```
1 REM **********************************      <132>
3 REM ** DEMO 1 - MIT NEUEN BEFEHLEN **       <146>
5 REM **********************************      <136>
10 :                                          <242>
100 OPEN 1,16:REM *OEFFNET DRUCKERKANAL       <098>
110 ←F:REM *FETTDRUCK AN*                     <135>
120 ←D:REM *DOPPELDRUCK AN*                   <045>
125 ←L10:REM *LINKEN RAND AUF 10 EINSTELLE
    N*                                        <013>
130 PRINT#1,"X";                              <213>
135 ←O:REM *POTENZIEREN AN*                   <132>
140 PRINT#1,"2";                              <085>
150 ←O0:REM *POTENZIEREN AUS*                 <187>
160 PRINT#1,"=256"                            <034>
```

© 64'er

Listing 2. Demo mit Druckerkurzbefehlen

Mit dem Drucker sprechen

Mit sinnvollen Kurzbefehlen erleichtert PROS V2.1 nicht nur die Arbeit mit dem Commodore 64 und dem Drucker MPS 802 ganz erheblich, es hält auch noch ein paar andere, ganz besondere Leckerbissen bereit. Und dabei arbeitet es sogar mit dem DOS V5.1 zusammen.

OPEN 1,4,7:CMD 1:LIST und dann PRINT #1:CLOSE 1, das ist es, was der gequälte Druckerbesitzer eingeben muß, bloß um ein paar eingegebene Basic-Programmzeilen schwarz auf weiß in den Händen halten zu können. Und da diese Eingabe nur im Direktmodus ohne weiteres möglich ist, fällt die endlose Tipperei jedesmal von neuem an.

Das hat jetzt ein Ende! Das PROS V2.1 nämlich bietet MPS 802-Benutzern für solch umfangreiche Befehlssequenzen sehr einfache Abkürzungen an. Aber auch Besitzer anderer Drucker, besonders des MPS 801, sollten ruhig weiterle-

sen. Denn wenn das Programm Anklang findet, ist eine baldige Entwicklung eines Zwillings für diese Drucker nicht auszuschließen.

Das Format der PROS-Befehle dürfte Floppy-Besitzern in ähnlicher Form bereits vom DOS V5.1 auf der Demo-Diskette bekannt sein. Sie alle bestehen lediglich aus dem Pfund-Symbol (oben rechts, die dritte Taste) und einem weiteren Zeichen. Das oben genannte Listen sähe mit dem PROS zum Beispiel ganz einfach so aus:

£ + (Pfund) und (Plus)
um die Ausgabe auf den Drucker zu lenken,
LIST
um das Programm zu listen und
£ — (Pfund) und (Minus)
um die Ausgabe wieder auf den Bildschirm zu holen.

Dem LIST-Befehl können Sie selbstverständlich auch Zeilenangaben beifügen, ganz wie Sie es bisher gewohnt waren.

Außer der normalen Ausgabeumleitung besteht noch die Möglichkeit, den Ausdruck zu formatieren. In der vorliegenden Version ist ein 40 Zeichen breites Ausgabefeld in Papiermitte eingestellt. Dieses Format ist zum Beispiel beim Ausdruck von Disk-Directories und disassemblierten Maschinenprogrammen sehr nützlich. Wie Sie die entsprechenden Parameter im PROS bereits bei der Eingabe ändern können, lesen Sie bitte weiter hinten.

Die Ausgabebefehle sind natürlich längst noch nicht alles, was das PROS zu bieten hat. Die Befehlsübersicht informiert Sie über die Einsatzmöglichkeiten der restlichen zehn PROS-Befehle.

Doch ich sprach eingangs von einigen besonderen Leckerbissen. Ich habe nämlich bei der Erstellung des Programms die Gelegenheit beim Schopfe ergriffen, noch einige andere Unbequemlichkeiten aus dem Weg zu räumen. So stört es zum Beispiel oft genug, daß beim Commodore 64 auf den Einbau einer Escape-Taste verzichtet wurde. Im PROS wird deshalb die Commodore-Taste zusätzlich mit dieser Funktion belegt, ohne dadurch in sonstigen Funktionen eingeschränkt zu werden.

Außerdem bekamen noch die beiden Funktionstasten F1 und F3 einen Auftrag zugewiesen: die Stop-Funktion. Wenn Sie in Zukunft einen ganz bestimmten Teil Ihres Basic-Programms suchen, sind Sie nicht mehr auf die CTRL-Taste angewiesen, um den LIST-Befehl zu verlangsamen. Sie brauchen nur F1 zu drücken und schon »schläft« der Computer. Erwecken Sie ihn dann mit der F3-Taste zu neuem Leben, so setzt er seine Arbeit genau dort fort, wo er sie zuvor unterbrochen hat.

Die Funktion dieser beiden Routinen beruht auf dem »Verbiegen« des Interrupt-Vektors in der Zeropage (Adresse 788/789). Bei Betätigung der Tasten RUN/STOP-RESTORE wird dieser Vektor jedoch automatisch wieder mit seinen Ursprungswerten versehen, das heißt die beiden Funktionen sind nicht mehr ohne weiteres verfügbar. Um Ihnen nun einen Neuaufruf des PROS (SYS 51555) zu ersparen, kann mit Hilfe von [Pfund] [Pfund] [Return] der entsprechende Programmteil neu aktiviert werden.

Ebenfalls neu aktiviert werden dabei der Kleinschriftmodus und die Repeat-Funktion für alle Tasten. Wollen Sie auf eine der beschriebenen Funktionen von vornherein verzichten, finden Sie auch hier die notwendigen Angaben zur Änderung weiter hinten.

Das Funktionsprinzip der eigentlichen PROS-Befehle basiert auf einer Veränderung des Vektors, der auf den Anfang jener Schleife zeigt, die einen Basic-Befehl holt und ausführt (Speicherstellen 776/777). Wann immer der Computer eine Basic-Zeile liest, im Programm oder im Direktmodus nach Drücken der Return-Taste, wird er nun zunächst prüfen, ob das jeweils gelesene Zeichen ein Pfundzeichen ist. Ist es das nicht, so wird der Prüfvorgang in der normalen Basic-Befehlstabelle fortgesetzt und gegebenenfalls der entsprechende Basic-Befehl ausgeführt. War das Zeichen hingegen tatsächlich ein »Pfund«, so wird jetzt in die PROS-Befehlstabelle gesprungen, um anhand des zweiten Zeichens den richtigen PROS-Befehl herauszusuchen.

Die Arbeitsgeschwindigkeit des Computers wird also kaum verringert, denn zunächst wird ja nur auf ein Zeichen mehr als üblich geprüft. Mit den zusätzlichen Befehlscodes wird erst verglichen, wenn feststeht, daß das PROS tatsächlich angesprochen werden soll.

Für die Abfragetechnik diente mir übrigens der Artikel »Maschinenprogramme auf Tastendruck« von P. Fleig in der 64'er, Ausgabe 12/84, Seite 80, als nützliche Vorlage.

Kommen wir nun zur Speicherbelegung des PROS V2.1. Es belegt die 648 Bytes von $C963 (dez. 51555) bis einschließlich $CBEA (dez. 52202). Daß ich genau diesen Speicherplatz gewählt habe, hat mehrere Gründe:

Erstens läßt das PROS so mehr als dreiviertel des Maschinenspeichers ($C000 - $CFFF) zusammenhängend frei.

Zweitens ist die Startadresse (51555) beim SYS-Aufruf denkbar leicht und entsprechend schnell einzutippen.

Und drittens beträgt der Abstand zum DOS 5.1, das sich ja bei Floppy-Benutzern großer Beliebtheit erfreut, nur genau 22 Bytes. So lassen sich das PROS, das DOS und sogar noch ein weiteres Programm, zum Beispiel das (verschobene) UNNEW (siehe D. Kossmann: Die Ebenen des Absturzes, 64'er, Ausgabe 11/84, Seite 92), mit Hilfe eines Maschinensprachemonitors hintereinander auf Diskette abspeichern und später auch zusammen wieder laden, ohne viel Speicherplatz zu verschwenden.

Um ein solches Kompaktprogramm noch besser nutzen zu können, enthält das PROS noch ein weiteres Extra: Die Speicherstellen $C96D-C96F (MSE-Listing: zweite Zeile, dritte bis fünfte Zahl) enthalten die Werte 4C, 70 und C9. Sie bewirken lediglich einen Sprung in die nächste Speicherstelle. Und doch liegt genau darin der Sinn dieser drei Bytes: Wenn Sie sie nämlich durch die Werte 20, 00 und CC ersetzen, wird beim Aufruf des PROS mit SYS 51555 das DOS (das Sie ja gleich mitgeladen haben) von hier aus mit aufgerufen. Sie sparen sich durch diese kleine Änderung also jedesmal den Aufruf des DOS.

Das PROS besteht aus zwei Teilen. Im ersten Abschnitt, das ist der, welcher beim Aufruf angesprungen wird, werden die erwähnten Vektoren in der notwendigen Weise geändert. Die ebenfalls bereits angesprochenen Interrupt-Routinen werden aktiviert und das Titelbild wird auf den Bildschirm gebracht.

Automatische Grundeinstellung

Auch mit diesem Titelbild hat es eine besondere Bewandtnis: Neben dem Namen des Programmierers, ich spreche von mir, enthält es eine Zeile, in der sich bereits einige PROS-Befehle, durch Doppelpunkte voneinander getrennt, befinden. Wenn sich der Computer nun nach dem Aufruf mit »READY« wieder meldet, befindet sich der Cursor genau in dieser Zeile. Sie brauchen also nur noch »RETURN« zu drücken und schon haben Sie ein paar notwendige Grundeinstellungen für den Drucker getroffen.

Diese Grundbefehle sind im einzelnen folgende:

1. Drucker in Grundstellung setzen
2. Fehlermeldungen des Druckers ermöglichen
3. zu druckende Zeilen pro Seite festlegen und
4. Seitenaufteilung anschalten.

Auch hier erhalten Sie weiter unten die notwendigen Informationen, die für eine Änderung beziehungsweise eine Ergänzung der Grundeinstellung notwendig sind.

Der zweite Abschnitt des PROS V2.1 beginnt bei $C974 (dez. 51572). Hier wird zunächst der Befehl geprüft und dann zu der entsprechenden Unterroutine gesprungen.

Das Programmlisting wurde mit Hilfe des MSE erstellt und wird mit ihm eingegeben.

Die Änderungen können mit einem Maschinensprachemonitor, zum Beispiel dem SMON, eingegeben werden.

Bedienungsanleitung

1. Aufruf

Das PROS V2.1 muß absolut geladen werden. Danach genügt die folgende Eingabe zum Aufruf:

NEW [Return]

SYS 51555 [Return]

Danach können die Grundeinstellungen durch nochmaliges Drücken der Return-Taste aktiviert werden.

2. Sonderfunktionen

Durch die Inbetriebnahme des PROS werden gleichzeitig einige besondere Funktionen eingeschaltet. Diese Funktionen werden durch die Tastenkombination RUN/STOP-RESTORE desaktiviert, können aber durch [Pfund] [Pfund] [Return] wieder aufgerufen werden.

Die Funktionen sind im einzelnen folgende:

Stop-Funktion

Ein Druck auf die Funktionstaste F1 läßt den Computer augenblicklich seine derzeitige Arbeit unterbrechen.

Durch Betätigen der Taste F3 wird der Arbeitsvorgang an der gleichen Stelle fortgesetzt, an der er unterbrochen wurde.

Escape-Taste

Die Commodore-Taste bekommt zusätzlich die Funktion einer Escape-Taste zugewiesen. Wird sie gedrückt, so verläßt der Computer den Quote-Modus (Hochkomma-Modus), das heißt, innerhalb von Anführungszeichen werden bei Benutzung der Cursortasten nicht mehr die Steuerzeichen ausgegeben.

Repeat-Funktion

Alle Tasten werden mit der Repeat-Funktion ausgestattet. Das einer Taste entsprechende Zeichen wird solange immer wieder ausgegeben, wie diese Taste gedrückt ist.

Groß-/Kleinschrift

Der Groß-/Kleinschriftmodus wird eingeschaltet.

3. Befehle

Alle PROS-Befehle beginnen mit dem Pfundzeichen. Dem folgt ein weiteres Zeichen und gegebenenfalls ein Parameter mit Werten zwischen 1 und 127.

Als PROS-Befehle werden anerkannt:

£S, <Parameter>

Festlegen des Zeilenabstandes. Drei Schritte entsprechen einer Punktbreite.

£P, <Parameter>

Festlegen der zu druckenden Zeilen pro Seite. Europäische Norm für Endlospapier: 66 Zeilen. Amerikanische Norm: 60 Zeilen.

Bei verändertem Zeilenabstand muß auch die Zahl der Zeilen pro Seite geändert werden, um eine saubere Seitenaufteilung zu erreichen. Es bieten sich dabei folgende Kombinationen an:

S24 - P102; S27 - P90; S32 - P75

S36 - P66 (Grundstellung)

S48 - P48; S54 - P42; S72 - P36

(bezogen auf die europäische Norm)

£(, £), [runde Klammer auf beziehungsweise zu]

Seitenaufteilung an- oder ausschalten. Bei angeschalteter Seitenaufteilung wird die festgelegte Zahl der Zeilen pro Seite gedruckt. Dann erfolgt ein Zeilenvorschub von sechs Zeilen und der Druck wird auf der nächsten Seite fortgesetzt.

£[,£] ,[eckige Klammer auf beziehungsweise zu]

Fehlermeldungen des Druckers ermöglichen beziehungsweise unterdrücken (Grundstellung).

£+, [Plus]

Ausgabe auf den Drucker lenken. Dieser Befehl entspricht in Basic der Befehlsfolge

OPEN 1,4,7:CMD 1

£F

Formatierte Ausgabe auf den Drucker lenken. In der Grundversion des PROS V2.1 ist ein 48 Zeichen breites Ausgabefeld in der Papiermitte, wie es zum Ausdruck von Disk-Direktories und disassemblierten Maschinenprogrammen nützlich ist.

Achtung: Soll der Ausdruck mit Hilfe eines Maschinensprachemonitors erfolgen, empfiehlt es sich, vorher die Ausgabe von Fehlermeldungen zu unterdrücken, da unter Umständen der Ausgabeabschluß nicht korrekt erfolgen kann. Auf die Arbeitsweise mit dem Drucker hat diese Fehlermeldung sonst keine Auswirkungen.

£—, [Minus]

Aufheben des durch £ + oder £ F erreichten Ausgabemodus. Nach diesem Befehl erfolgt die Datenausgabe wieder auf dem Bildschirm.

£ #, (Doppelkreuz)

Wiederherstellung der Druckergrundstellung (Einschaltzustand).

££ [Pfund]

Wiederaktivieren der PROS-Sonderfunktionen.

£Q

Verlassen des PROS V2.1. Nach Eingabe dieses Befehls sind alle PROS-Befehle außer Betrieb. Um wieder mit dem Programm arbeiten zu können, muß es mit SYS 51555 neu aufgerufen werden.

Die Sonderfunktionen werden durch diesen Befehl nicht aufgehoben.

Änderungsvorschläge

1. PROS-Befehlscode [Pfund] ändern

Sollen die PROS-Befehle mit einem anderen Zeichen als dem »Pfund« beginnen, so muß der ASCII-Code des gewünschten Zeichens in der Speicherstelle $C978 (dez. 51576) stehen. In der Normalversion lautet dieser Wert $5C (dez. 92).

2. Änderung beziehungsweise Ergänzung der Grundbefehle in der Titelseite

Die Grundbefehle befinden sich in den Speicherstellen $CA58 - $CA7c (dez. 51800 - 51836). Die Befehle müssen als hexadezimale ASCII-Codes eingegeben werden. Der Code für das Pfundzeichen lautet $5C (92).

3. Ausschalten der Kleinschrift

Um beim PROS-Aufruf den Großschriftmodus beizubehalten, müssen die Bytes $CA90 - $CA95 mit dem Wert $EA (Befehlscode für NOP) aufgefüllt werden.

Um auch im Ausgabemodus (PROS-Befehl [Pfund] und [Plus]) im Großschriftmodus zu bleiben, muß die Speicherstelle $CBA5 (dez. 52133) statt $07 den Wert $00 erhalten.

4. Ausschalten der Repeatfunktion für alle Tasten

Um diesen Effekt zu vermeiden, sind die Bytes $CA8B - $CA8F mit dem Wert $EA aufzufüllen.

5. Ausschalten der Stop-Funktion und der Escape-Taste

Die entsprechenden Routinen lassen sich umgehen, indem in den Speicherstellen $C9E7 (dez. 51687) und

$CB9F (dez. 52127) jeweils der Wert $7F durch $8B ersetzt wird.

Der Bereich $CA7F - $CA8A (dez. 51839 - 51850) wird dadurch übersprungen und kann zum Beispiel für individuelle PROS-Erweiterungen als Speicherplatz dienen.

6. Änderung der Tastenbelegung der Stop-Funktion

Um die Tastenbelegung für die Stop-Funktion (normalerweise F1 und F3) zu ändern, müssen die Tastaturcodes der gewünschten Tasten in die Speicherstellen $CA9A (statt F1-Taste) und $CAB3 (statt F3-Taste) geschrieben werden. Die Dezimalwerte der Adresse: 51866 und 51891.

7. Escape-Taste verlegen

In der PROS V2.1 Normalversion versieht die Commodore-Taste zusätzlich den Dienst einer Escape-Taste. Ersatzweise können auch die Taste [CTRL] oder [Pfeil nach links] mit dieser Aufgabe betraut werden. Dazu muß die Stelle $CABB den Wert $FB (dez. 251) beziehungsweise $FD (dez. 253) bekommen.

8. Änderung der Gerätenummer

Das PROS V2.1 kann der Gerätenummer des benutzten Druckers angepaßt werden, indem der neue Wert in die Stelle $CACE (dez. 51918) geschrieben wird.

9. Verändern des formatierten Ausgabefeldes

Das in der Normalversion festgelegte formatierte Ausgabefeld (40 Zeichen breit in Papiermitte) kann wie folgt verändert werden:

Die gewünschte Breite des linken Randes ist in der Speicherstelle $CBC8 (dez. 52168), die des Ausgabefeldes in $CBD2 (dez. 52178) abzulegen.

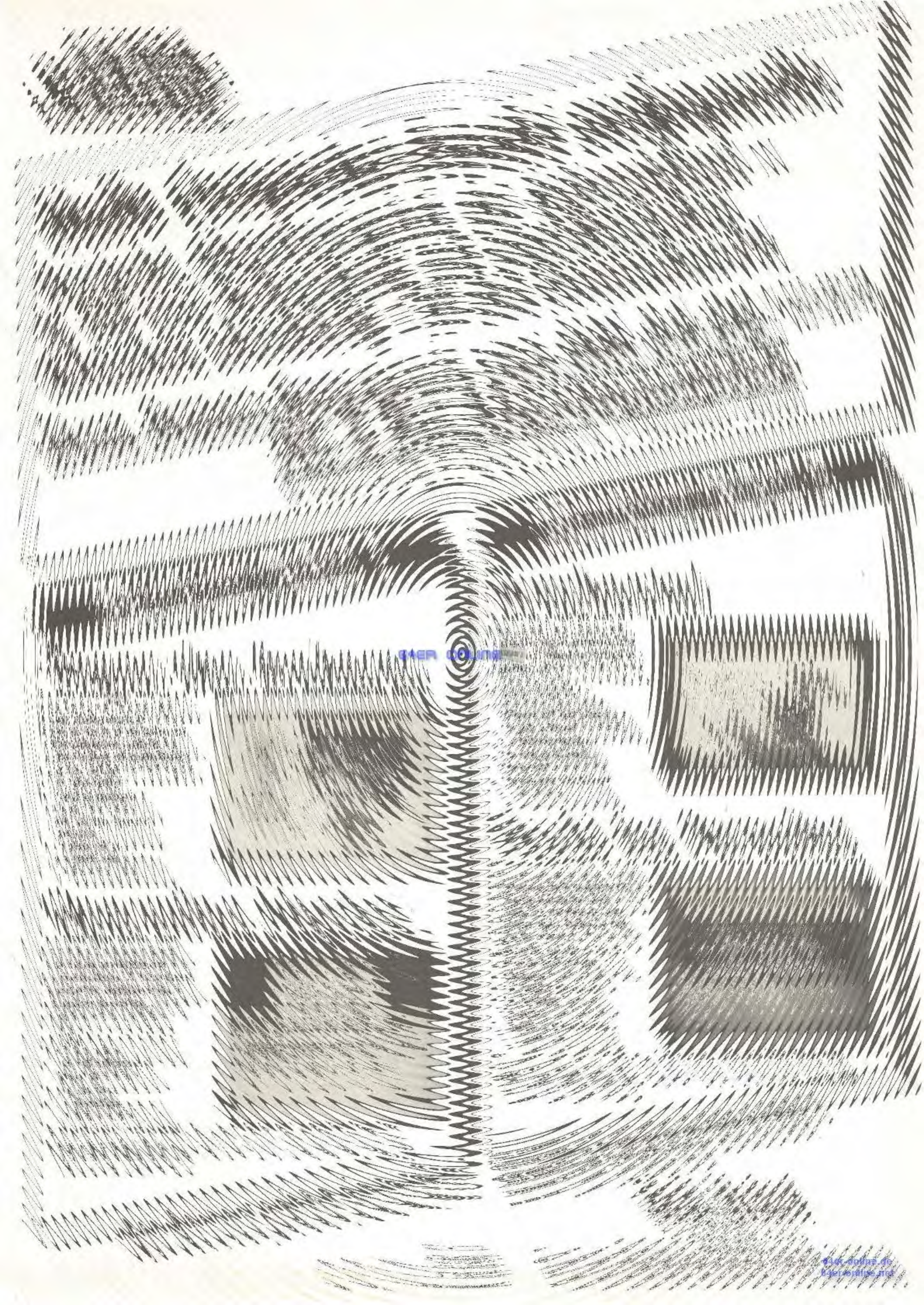
Hinweis: Auf der Leserservice-Diskette befindet sich einmal das PROS V2.1 selbst und zusätzlich mit integriertem DOS V5.1 und dem Programm UNNEW. Aufruf von UNNEW: SYS 53166.

(Peter T. Schmidt/gk)

```
programm : pros v2.1        c963 cbeb
-------------------------------------

c963 : a9 74 8d 08 03 a9 c9 8d   6a
c96b : 09 03 4c 70 c9 20 d9 c9   b0
c973 : 60 20 73 00 c9 5c f0 06   0f
c97b : 20 79 00 4c e7 a7 20 73   05
c983 : 00 4c 8a c9 4c ae a7 a2   a3
c98b : 00 dd ac c9 f0 0a e8 e8   b3
c993 : e8 e0 27 f0 e3 4c 8c c9   3a
c99b : bd ad c9 8d aa c9 bd ae   a0
c9a3 : c9 8d ab c9 6c aa c9 56   47
c9ab : cb 53 23 cb 50 2b cb 5b   a6
c9b3 : 33 cb 5d 38 cb ae 4b cb   21
c9bb : 28 56 cb 29 61 cb 51 6c   b9
c9c3 : cb 23 93 cb 5c 9e cb aa   bd
c9cb : a4 cb ab b4 cb 46 bf cb   5c
c9d3 : 20 73 00 4c 87 c9 a2 00   88
c9db : bd e9 c9 20 d2 ff e8 e0   96
c9e3 : 96 d0 f5 4c 7f ca 11 20   bb
c9eb : 20 20 20 20 20 20 20 20   eb
c9f3 : 20 2a 2a 2a 2a 20 d0 52   84
c9fb : 4f 53 20 20 d6 32 2e 31   1a
ca03 : 20 2a 2a 2a 2a 0d 11 20   98
ca0b : 20 20 20 20 20 20 20 d0   6d
ca13 : 52 49 4e 54 45 52 20 cf   2f
ca1b : 50 45 52 41 54 49 4e 47   22
ca23 : 20 d3 59 53 54 45 4d 0d   ac
ca2b : 11 20 20 20 20 28 43 29   fb
ca33 : 20 42 59 20 d0 45 54 45   e2
ca3b : 52 20 d4 2e 20 d3 43 48   d6
ca43 : 4d 49 44 54 2c 31 39 38   72
ca4b : 34 2f 38 35 0d 11 11 11   8b
ca53 : 11 11 11 91 91 20 20 20   3e
ca5b : 20 5c 23 20 3a 20 5c 5b   43
ca63 : 20 3a 20 5c 50 36 36 20   04
ca6b : 3a 20 5c 28 20 20 20 20   95
ca73 : 20 20 20 20 20 20 20 20   73
ca7b : 20 20 91 91 78 a9 96 8d   8c
ca83 : 14 03 a9 ca 8d 15 03 58   1b
ca8b : a9 ff 8d 8a 02 a9 17 8d   cd
ca93 : 18 d0 60 48 a5 cb c9 04   1c
ca9b : f0 04 68 4c b8 ca a9 ca   4f
caa3 : 48 a9 b0 48 08 48 8a 48   72
caab : 98 48 4c b8 ca a5 cb c9   2e
cab3 : 05 d0 eb f0 e5 a5 91 c9   9f
cabb : df d0 06 a9 00 85 d4 85   44
cac3 : d8 4c 31 ea 00 00 88 ad   e8
cacb : c9 ca a2 04 ac c7 ca 20   97
cad3 : ba ff a9 00 20 bd ff 20   27
cadb : c0 ff ae c9 ca 20 c9 ff   54
cae3 : 60 ad c8 ca 20 d2 ff a9   91
caeb : 0d 20 d2 ff 60 20 cc ff   f7
caf3 : ad c9 ca 20 c3 ff a9 00   1e
cafb : 8d c7 ca 8d c8 ca a9 88   6b
cb03 : 8d c9 ca 60 20 79 00 20   42
cb0b : 9b b7 8e c8 ca 20 19 cb   e8
cb13 : 20 79 00 4c 87 c9 20 ca   56
cb1b : ca 20 e4 ca 20 f0 ca 60   fd
cb23 : a9 06 8d c7 ca 4c 07 cb   ee
cb2b : a9 03 8d c7 ca 4c 07 cb   75
cb33 : a9 04 4c 3d cb a9 09 4c   60
cb3b : 3d cb 8d c7 ca a9 0a 8d   f7
cb43 : c8 ca 20 19 cb 4c d3 c9   9e
cb4b : a9 0c 8d c8 ca 20 19 cb   20
cb53 : 4c d3 c9 a9 93 8d c8 ca   8f
cb5b : 20 19 cb 4c d3 c9 a9 13   dd
cb63 : 8d c8 ca 20 19 cb 4c d3   d4
cb6b : c9 a9 e4 8d 08 03 a9 a7   82
cb73 : 8d 09 03 a2 00 bd 86 cb   3a
cb7b : 20 d2 ff e8 e0 0d d0 f5   c7
cb83 : 4c d3 c9 0d 20 47 4f 4f   e5
cb8b : 44 20 42 59 45 20 21 0d   8f
cb93 : a9 0a 8d c7 ca 20 19 cb   47
cb9b : 4c d3 c9 20 7f ca 4c d3   6e
cba3 : c9 a9 07 8d c7 ca a9 89   41
cbab : 8d c9 ca 20 ca ca 4c d3   af
cbb3 : c9 a9 89 8d c9 ca 20 f0   ba
cbbb : ca 4c d3 c9 a9 02 8d c7   4a
cbc3 : ca 20 ca ca a2 14 a9 20   5b
cbcb : 20 d2 ff ca d0 f8 a2 28   5d
cbd3 : a9 41 20 d2 ff ca d0 f8   0b
cbdb : a9 0d 20 d2 ff 20 f0 ca   c8
cbe3 : a9 01 8d c7 ca 4c a9 cb   b6
```

Listing PROS V2.1. Bitte beachten Sie die Eingabehilfe auf Seite 8.



Sprite+Grafik-Basic

Diese für den C 64 geschriebene Basic-Erweiterung wurde speziell für das Programmieren von Sprites und hochauflösender Grafik entwickelt, enthält jedoch auch zusätzliche, nicht grafikbezogene Befehle.

Sprite+Grafik-Basic wurde für den Commodore 64 ohne besondere Peripherie geschrieben. Das Programm wird mit LOAD geladen und mit RUN gestartet.

Das für den Commodore 64 geschriebene Sprite+Grafik-Basic ist eine Basic-Erweiterung, die speziell das Programmieren von Sprites und hochauflösender Grafik unterstützt. Selbstverständlich werden auch Befehle zum Editieren der Programme und die nötigsten strukturierenden Befehle zur Verfügung gestellt.

Sprite+Grafik-Basic erweitert das bescheidene V2-Basic um mehr als **100** Befehle. Davon sind 32 Befehle zur Programmierung der Sprites vorgesehen.

Außer den üblichen Parametern der Spriteprogrammierung können Sie die Sprites spiegeln, drehen und noch vieles mehr. Durch die vielseitigen Befehle zur Definition der Spritematrix kann selbst der Anfänger ohne einen zusätzlichen Spriteeditor hervorragende Sprites erstellen. Die Spritematrixblöcke — es stehen maximal 144 davon zur Verfügung — benötigen natürlich keinen Basic-Programmspeicher.

Auch zum Erstellen von ansprechenden Bildschirmmasken bietet Sprite+Grafik-Basic 14 neue Befehle. Es ist zum Beispiel möglich, den Bildschirm in alle vier Richtungen zu scrollen oder bestimmte Flächen mit Zeichen zu füllen.

Für die Programmierung von hochauflösender Grafik und Multicolorgrafik stehen natürlich die Befehle, um Punkte und Linien zu setzen zur Verfügung. Darüber hinaus können Ellipsen, Ellipsenbogen, Radien, Rechtecke und Blöcke mit einfachen Befehlen definiert werden. Eine der Besonderheiten von Sprite+Grafik-Basic ist der GPRINT-Befehl, der es ermöglicht — annähernd so komfortabel wie der PRINT-Befehl der Blockgrafik — Text in die Grafik einzubauen. Bei jedem dieser Befehle kann natürlich gewählt werden, ob ein Punkt gesetzt, gelöscht oder invertiert werden soll.

Nach dem Ausschalten der Grafik und dem Wiedereinschalten bleibt diese vollständig erhalten. Hierdurch kann sie auch im ausgeschalteten Zustand durch zwei Befehle abgespeichert oder geladen werden.

Über einen SYS-Befehl und wenige POKEs läßt sich sogar ein Textfenster in die Grafik einblenden.

Insgesamt bietet Sprite+Grafik-Basic 16 High-Resolution-Befehle. Diese Grafikmöglichkeiten werden durch das Erzeugen von blinkenden Zeichen oder durch Blinken des Rahmens zusätzlich verstärkt. Die Programmierung von Sprites und Grafik läßt also kaum Wünsche offen.

Damit diese Sprite+Grafik-Programmierung möglichst einfach und übersichtlich wird, gibt es in Sprite+Grafik-Basic knapp zehn Befehle zur Strukturierung der Programme. Dies sind alle üblichen Befehle wie zum Beispiel REPEAT ... UNTIL, CASE ... THEN ... ELSE und den mit Labeln arbeitenden Sprungbefehlen. Außerdem existiert ein Befehl, mit dem Sie die letzte IF-Bedingung noch einmal abfragen können.

Um all diese Befehle platzsparend und einfach in ein Programm einbauen zu können, stehen etwa 20 weitere Befehle zur Verfügung. Man kann damit sofort editieren oder auch nach Zeichenketten und Befehlen im Programm suchen. Es ist ebenfalls möglich, ein Programm zeilenweise abzuspeichern oder ein Programm nachzuladen.

Zur Vereinfachung der Programmierung können die Funktionstasten mit beliebigen Zeichenketten belegt werden.

Falls ein Programm nicht funktioniert, findet man mit dem TRACE-Modus sehr leicht den Fehler. Dieser Modus zeigt immer die aktuelle Zeile an — natürlich auch in High-Resolution-Grafik. Um Fehler bei den neuen Befehlen besser zu verstehen, sind noch einige neue Fehlermeldungen implementiert. Außerdem wird die fehlerhafte Zeile angezeigt und die fehlerhafte Stelle markiert.

Blockgrafikbefehle (40x25 Auflösung)

Syntax : **AT x , y , string**
Parameter : x — x-Koordinate (0 bis 39)
y — y-Koordinate (0 bis 24)
string — Zeichenkette
Funktion : Gibt einen String an der angegebenen Bildschirmposition aus.

Syntax : **SET c , f , x , y**
Parameter : x — x-Koordinate (0 bis 39)
y — y-Koordinate (0 bis 24)
c — Bildschirmcode des Zeichens (0 bis 255)
f — Farbe (0 bis 15)
Funktion : Gibt ein Zeichen an der angegebenen Bildschirmposition aus.

Syntax : **FSET f , x , y**
Parameter : x — x-Koordinate (0 bis 39)
y — y-Koordinate (0 bis 24)
f — Farbe (0 bis 15)
Funktion : Setzt die Farbe an der angegebenen Bildschirmposition.

Syntax : **ZSET c , x , y**
Parameter : x — x-Koordinate (0 bis 39)
y — y-Koordinate (0 bis 24)
c — Bildschirmcode des Zeichens (0 bis 255)
Funktion : Gibt ein Zeichen an der angegebenen Bildschirmposition aus.

Syntax : **COLOUR rf , hf**
Parameter : rf — Rahmenfarbe (0 bis 15)
hf — Hintergrundfarbe (0 bis 15)
Funktion : Setzt die Farben.

Syntax : **CURSOR f**
Parameter : f — Farbe (0 bis 15)
Funktion : Setzt die Zeichenfarbe.

Syntax : **FILL x1 , y1 , x2 , y2 , c , f**
Parameter : x1 — x-Koordinate der linken oberen Ecke (0 bis 39)
y1 — y-Koordinate der linken oberen Ecke (0 bis 24)
x2 — x-Koordinate der rechten unteren Ecke (0 bis 39)
y2 — y-Koordinate der rechten unteren Ecke (0 bis 24)

c — Bildschirmcode des Zeichens (0 bis 255)
f — Farbe (0 bis 15)
Funktion : Ausfüllen einer Fläche mit Zeichen in einer bestimmten Farbe.

Syntax : **ZFILL x1 , y1 , x2 , y2 , c**
Parameter : wie »FILL«
Funktion : Ausfüllen einer Fläche mit bestimmten Zeichen.

Syntax : **INV x1 , y1 , x2 , y2**
Parameter : wie »FILL«
Funktion : Invertiert alle Zeichen im angegebenen Bildschirmbereich.

Syntax : **FFILL x1 , y1 , x2 , y2**
Parameter : wie »FILL«
Funktion : Einfärben einer Fläche mit der angegebenen Farbe.

Syntax : **UP/DOWN/LEFT/RIGHT x1 , y1 , x2 , y2**
Parameter : wie »FILL«
Funktion : Scrollt den angegebenen Bereich.
Beispiel : LEFT 10,10,20,20
Der angegebene Bereich wird nach links gescrollt.

Flashbefehle

Syntax : **VEC**
Parameter : keine
Funktion : Schaltet auf die neue Interruptroutine um.

Syntax : **OLDVEC**
Parameter : keine
Funktion : Schaltet den alten Interrupt wieder ein.

Syntax : **B/G/BG FLASH f1 , f2 , speed**
Parameter : f1 — Farbe 1 (0 bis 15)
f2 — Farbe 2 (0 bis 15)
speed — Blinkgeschwindigkeit in 1/10 Sekunden (1 bis 255)
Funktion : Rahmen und/oder Hintergrund wechseln ständig zwischen den angegebenen Farben.
Beispiele : BFLASH 0,1,1
Rahmen wechselt jede 1/10 Sekunde zwischen Schwarz und Weiß.
BGFLASH 0,1,2
Rahmen und Hintergrund wechseln jede zweite 1/10 Sekunde zwischen Schwarz und Weiß.

Syntax : **FOFF**
Parameter : keine
Funktion : B/G/BG FLASH wird ausgeschaltet.

Syntax : **FLASH f1 , f2 , f3**
Parameter : f1,f2,f3 — Farben (0 bis 15)
Funktion : Alle Zeichen, die die Farbe f1, f2 oder f3 haben, werden ständig invertiert.

Syntax : **OFF**
Parameter : keine
Funktion : FLASH wird ausgeschaltet.

Syntax : **SCLR adresse**
Parameter : adresse — Anfangsadresse des Spriteblocks
32768+64*Spriteblock
Funktion : Löscht einen Spriteblock.
Es stehen Block 16 bis 31, und wenn Sie keine hochauflösende Grafik bearbeiten, auch noch Block 128 bis 255 zur Verfügung.

Syntax : **SRVS adresse**
Parameter : wie »SCLR«
Funktion : Invertiert einen Spriteblock.

Syntax : **BMOVE adresse1 , adresse2**
Parameter : wie »SCLR«
Funktion : Kopiert Spriteblock 1 in Spriteblock 2.

Syntax : **SON sn**
Parameter : sn — Spritenummer (0 bis 7)
Funktion : Schaltet ein Sprite ein.

Syntax : **SOFF sn**
Parameter : sn — Spritenummer (0 bis 7)
Funktion : Schaltet ein Sprite aus.

Syntax : **SBLOCK sn , block**
Parameter : sn — Spritenummer (0 bis 7)
block — Spriteblock (0 bis 255)
Funktion : Gibt an, aus welchem Block das Sprite seine Matrix erhält.

Syntax : **SMODE sn , modus**
Parameter : sn — Spritenummer (0 bis 7)
modus = 0 — Multicolorsprite
= 1 — Normalsprite
Funktion : Einem Sprite wird sein Darstellungsmodus zugewiesen.

Syntax : **SCOLOUR sn , f1**
Parameter : sn — Spritenummer (0 bis 7)
f1 — Farbe (0 bis 15)
Funktion : Einem Sprite wird die Farbe zugewiesen.

Syntax : **MCOLOUR f2 , f3**
Parameter : f2,f3 — Farben (0 bis 15)
Funktion : Allen Multicolorsprites werden die restlichen Farben zugewiesen.

Syntax : **SPRIOR sn , modus**
Parameter : sn — Spritenummer (0 bis 7)
modus = 0 — Priorität
= 1 — keine Priorität
Funktion : Es wird festgelegt, ob das Sprite vor dem Hintergrund die Priorität hat oder nicht.

Syntax : **X/Y EXP sn , modus**
Parameter : sn — Spritenummer (0 bis 7)
modus = 0 — Normalgröße
= 1 — expandiert
Funktion : Es wird festgelegt, ob das Sprite normal oder doppelt so breit dargestellt wird.
Beispiel : XEXP 4,1
Sprite 4 wird in x-Richtung doppelt so breit dargestellt.

Syntax : **SSET sn , x , y**
Parameter : sn — Spritenummer (0 bis 7)
x — x-Koordinate (0 bis 511)
y — y-Koordinate (0 bis 255)
Funktion : Setzt das angegebene Sprite an die x/y-Position.

Syntax : **S UP/DOWN/LEFT/RIGHT sn , sw**
Parameter : sn — Spritenummer (0 bis 7)
sw — Schrittweite (0 bis 255)
Funktion : Bewegt das Sprite um sw Pixels in die angegebene Richtung.
Beispiele : SUP 2,100
Bewegt Sprite 2 um 100 Bildpunkte nach oben.
SRIGHT 5,20
Bewegt Sprite 5 um 20 Bildpunkte nach rechts.

Syntax : **SPLOT ad , x , y , modus , zm**
Parameter : ad — wie »SCLR«
x — x-Koordinate
High-Resolution : (0 bis 23)
Multicolor : (0 bis 11)
y — y-Koordinate (0 bis 20)
modus = 0 — Normalsprite
= 1 — Multicolorsprite
zm — Zeichenmodus
High-Resolution: 0 löschen
1 setzen
2 invertieren
Multicolor: 0 löschen
1 f1 setzen
2 f2 setzen
3 f3 setzen
Funktion : Setzt/löscht/invertiert einen Punkt in einem Speicherblock.

Syntax : **DESIGN ad , y , modus , string**
Parameter : ad — wie »SCLR«
y — y-Koordinate (0 bis 20)
modus — wie »SPLOT«
string — Spritematrixzeile
High-Resolution: 24 Zeichen
A setzen
. löschen
B keine Veränderung
I invertieren
Multicolor: 12 Zeichen
A f1 setzen
B f2 setzen
C f3 setzen
D keine Veränderung
I invertieren
. löschen
Funktion : Definiert die Spritematrixzeile eines Spriteblockes.
Beispiele : DESIGN 40960,0,1,"ABACAI..IIBC"
Definiert die Spritezeile 0 des Blockes 128 mit "ABACAI..ABI."
A$="AAAABBBBAAAAIIII..ABI."
DESIGN 40960,6,0,A$
Definiert die Spritezeile 5 des Blockes 128 mit dem String A$.

Syntax : **SCORD sn , v1 , v2**
Parameter : sn — Spritenummer (0 bis 7)
v1,v2 — nichtindizierte, reelle Variable
Funktion : Übergibt die x-Koordinate des angegebenen Sprites in die Variable v1 und die y-Koordinate in v2.

Syntax : **SMOVE sn , x1 , y1 , x2 , y2 , sw , s**
Parameter : sn — Spritenummer (0 bis 7)
x1,y1 — Koordinaten der Ausgangsposition, wie »SSET«
x2,y2 — Koordinaten des Zielpunkts, wie »SSET«
sw — Schrittweite (1 bis 255)
s — Geschwindigkeit (1 bis 255)
Funktion : Bewegt ein Sprite von der Ausgangsposition zum Zielpunkt in sw Teilschritten.

Syntax : **MOVETO sn , x , y , sw , s**
Parameter : wie »SMOVE«
Funktion : Bewegt ein Sprite von seiner derzeitigen Position zur angegebenen x/y-Koordinate.

Syntax : **R RIGHT/LEFT/UP/DOWN ad**
Parameter : wie »SCLR«
Funktion : Scrollt die Spritematrix des angegebenen Blocks.
Beispiel : RUP 40960
Scrollt die Matrix des Blocks 128 nach oben.

Syntax : **XMIR modus , ad**
Parameter : modus = 0 — Normalsprite
= 1 — Multicolorsprite
ad — wie »SCLR«
Funktion : Spiegelt die Matrix des angegebenen Spriteblocks in x-Richtung.

Syntax : **YMIR ad**
Parameter : wie »SCLR«
Funktion : Spiegelt die Matrix des angegebenen Spriteblocks in y-Richtung.

Syntax : **TURN modus , ad**
Parameter : wie »SCLR« und »XMIR«
Funktion : Dreht die Matrix des angegebenen Spriteblocks um 180 Grad.

Syntax : **ROTR ad**
Parameter : wie »SCLR«
Funktion : Dreht die Matrix des angegebenen Spriteblocks um 90 Grad nach rechts.
Hinweis : Arbeitet nicht bei Multicolorsprites.

Syntax : **ROTL ad**
Parameter : wie »SCLR«
Funktion : Dreht die Matrix des angegebenen Spriteblocks um 90 Grad nach links.
Hinweis : Arbeitet nicht bei Multicolorsprites.

Syntax : **DETECT modus**
Parameter : modus = 0 — löschen des Sprite-Sprite-Kollisionsregisters
= 1 — löschen des Sprite-Hintergrund-Kollisionsregisters
Funktion : Setzt die Kollisionsregister zurück.

Befehle für die hochauflösende Grafik

Hinweise : **Koordinatenangaben**
High-Resolution: x-Koordinaten (0 bis 319)
y-Koordinaten (0 bis 199)
Multicolor: x-Koordinaten (0 bis 159)
y-Koordinaten (0 bis 199)

Koordinatenangaben, die über den Bereich hinausgehen, werden ignoriert.

Zeichenmodus zm
High-
Resolution: 0 — löschen
1 — setzen
2 — invertieren
Multicolor: 0 — löschen
1 — Farbe 1 setzen
2 — Farbe 2 setzen
3 — Farbe 3 setzen
4 — invertieren

Syntax : **HRG pf , hf**
Parameter : pf — Punktfarbe (0 bis 15)
hf — Hintergrundfarbe (0 bis 15)
Funktion : Schaltet die hochauflösende Grafik ein und legt die Farben fest.

Syntax : **NRM**
Parameter : keine
Funktion : Schaltet wieder auf normale Blockgrafik um. Dabei wird der Bildschirm gelöscht und die Hintergrund- und Rahmenfarbe wie bei RUN/STOP-RESTORE gesetzt.

Syntax : **CLS**
Parameter : keine
Funktion : Löscht den Grafikbildschirm und setzt die Farben, wie bei »HRG« festgelegt.

Syntax : **MULTI f1 , f2 , f3**
Parameter : f1,f2,f3 — Farben (0 bis 15)
Funktion : Schaltet Multicolormodus ein und setzt die drei Farben.

Syntax : **MODE zm**
Parameter : zm — Zeichenmodus
Funktion : Stellt den Zeichenmodus ein. Er gilt für alle nachfolgenden Grafikbefehle.

Syntax : High- **GCOLOUR pf, hf**
Resolution: Parameter wie bei »HRG«

Multicolor: **GCOLOUR f1 , f2 , f3**
Parameter wie »MULTI«
Funktion : Setzt die Farben für alle nachfolgenden Grafikbefehle.

Syntax : **PLOT x , y**
Parameter : x,y — Koordinaten
Funktion : Setzt einen Punkt.

Syntax : **DRAW x , y**
Parameter : x,y — Koordinaten
Funktion : Zieht eine Linie vom zuletzt gesetzten Punkt zur neuen Koordinate.

Syntax : **LINE x1 , y1 , x2 , y2**
Parameter : x1,y1,x2,y2 — Koordinaten
Funktion : Zieht eine Linie von der Koordinate x1/y1 zur Koordinate x2/y2.

Syntax : **REC x1 , y1 , x2 , y2**
Parameter : x1,y1 — Koordinaten der linken oberen Ecke
x2,y2 — Koordinaten der rechten unteren Ecke
Funktion : Zeichnet ein Rechteck mit den angegebenen Eckpunkten.

Syntax : **BLOCK x1, y1 , x2 , y2**
Parameter : wie »REC«
Funktion : Zeichnet ein ausgefülltes Rechteck mit den angegebenen Eckpunkten.

Syntax : **ANGL x , y , rx , ry , w**
Parameter : x,y — Mittelpunktskoordinaten
rx — Radius in x-Richtung
ry — Radius in y-Richtung
w — Winkel (0 bis 360)
Funktion : Zeichnet den Radius einer Ellipse.

Syntax : **ARC x , y , rx , ry , sw , ew , a**
Parameter : x,y — Mittelpunktskoordinaten
rx,ry — wie »ANGL«
sw — Startwinkel (0 bis 360)
ew — Endwinkel (0 bis 360)
a — Winkelabstand der berechneten Kreispunkte (0 bis 360)
Funktion : Zeichnet einen Ellipsenbogen.

Syntax : **GPRINT string**
Parameter : string — Zeichenkette
Funktion : Gibt den in string enthaltenen Text auf dem Grafikbildschirm aus. Es kann Schriftgröße, Zeichensatz, Farbe und Cursorsteuerung durch Steuerzeichen beeinflußt werden. Der Cursor ist allerdings nie sichtbar. Er steht ständig an der Position des zuletzt ausgegebenen Grafikpunktes. Die aktuellen Schriftparameter können aus Speicherstellen ausgelesen werden (Tabelle 1).

Syntax : **GSAVE gn , name**
Parameter : gn — Gerätenummer
name — Namensstring
Funktion : Speichert die hochauflösende Grafik ab. Dies erfolgt in drei Blöcken.
Hinweis : Bei allen LOAD/SAVE-Vorgängen werden die Schriftparameter von GPRINT zerstört. Das Blinken der Floppy-LED ist normal.

Syntax : **GLOAD gn , name**
Parameter : wie »GSAVE«
Funktion : Lädt eine hochauflösende Grafik.

Editorbefehle

Syntax : **FKEY fn , string**
Parameter : fn — Funktionstastennummer (1 bis 12)
string — Zeichenkette mit maximal 20 Zeichen
Funktion : Weist einer Funktionstaste eine Zeichenkette zu. Die Funktionstasten 9 bis 12 werden über die Commodore-Taste erreicht. Der neue Interrupt muß mit »VEC« aktiviert werden.

Syntax : **DISP**
Parameter : keine
Funktion : Zeigt die Funktionstastenbelegung an.

Syntax : **PAGE zn**
Parameter : zn — Zeilennummer
Funktion : Listet die 10 Basic-Zeilen nach zn.

Syntax : **MOVE**
Parameter : keine
Funktion : Die nachfolgenden 10 Zeilen werden ausgegeben.

Syntax : **BACK**
Parameter : keine
Funktion : Gibt die 10 Zeilen vor den zuletzt angezeigten aus.
Hinweis : Nach den Befehlen PAGE, MOVE und BACK befinden Sie sich im Direktmodus und können die Zeilen editieren.

Syntax : **OLD**
Parameter : keine
Funktion : Ein durch NEW gelöschtes Programm wird wieder aktiviert.

Syntax : **DUMP**
Parameter : keine
Funktion : Gibt die Inhalte aller Variablen aus und zeigt die Funktionsnamen selbstdefinierter Funktionen an. Sie werden durch »!« gekennzeichnet.

Syntax : **ARRAY v**
Parameter : v — Name des Feldes
Funktion : Gibt den Inhalt aller Elemente des angegebenen Feldes aus.

Syntax : **MEM**
Parameter : keine
Funktion : Zeigt die momentane Speicherbelegung an.

Syntax : **MERGE name , gn**
Parameter : wie »GSAVE«
Funktion : Hängt ein Programm an das im Speicher stehende an. Dabei sollten die Zeilennummern des Programms im Speicher kleiner sein als die des geladenen.

Syntax : **ZSAVE gn , name , z1 , z2**
Parameter : gn,name — wie »GSAVE«
z1,z2 — Zeilennummern
Funktion : Speichert das Programm von Zeilennummer z1 bis z2 ab.

Syntax : **ZLOAD z , name , gn**
Parameter : gn,name — wie »GSAVE«
z — Zeilennummer
Funktion : Lädt ein Programm und fügt es ab der Zeile z an. Die Zeilen, die bisher auf z folgten, gehen verloren.

Syntax : **DEL z1 , z2**
Parameter : z1,z2 — Zeilennummern
Funktion : Löscht das Programm von Zeilennummer z1 bis z2.

Syntax : **HELP**
Parameter : keine
Funktion : Gibt alle Basic V 2.0-Befehle aus.

Syntax : **@HELP**
Parameter : keine
Funktion : Gibt alle Sprite+Grafik-Basicbefehle aus.

Syntax : **FIND Zeichenkette**
Parameter : zeichenkette — beliebige Zeichen
Funktion : Gibt die Zeilennummern der Zeilen, in denen die Zeichenkette enthalten ist, aus.
Beispiel : FINDGOTO
Die Zeilennummern der Zeilen, in denen ein GOTO steht, werden ausgegeben.

Syntax : **AUTO zn , a**
Parameter : zn — Zeilennummer
a — Abstand
Funktion : Automatische Zeilenvorgabe. Ist die vorgegebene Zeile schon im Programm enthalten, färbt sich der Rahmen rot.

Syntax : **TRACE**
Parameter : keine
Funktion : In der obersten Bildschirmzeile wird die aktuelle Programmzeile angezeigt und der aktuelle Befehl negativ gekennzeichnet. Dieser Befehl wird erst abgearbeitet, wenn eine Taste gedrückt wurde.

Syntax : **RETRACE**
Parameter : keine
Funktion : Schaltet den TRACE-Modus aus.

Syntax : **ERROR**
Parameter : keine
Funktion : Gibt den zuletzt aufgetretenen Fehler und die dazugehörige Zeile aus.

Syntax : **COLD**
Parameter : keine
Funktion : Startet das Sprite+Grafik-Basic neu.

Strukturierende Befehle

Syntax : **REPEAT:...:UNTIL Bedingung**
Parameter : Bedingung — Verzweigungsbedingung
Funktion : Führt die auf REPEAT folgenden Anweisungen so lange durch, bis die Bedingung nach UNTIL zutrifft.

Syntax : **CASE Bedingung THEN....:ELSE....**
Parameter : bedingung — Verzweigungsbedingung
Funktion : Ist die Bedingung erfüllt, werden die Befehle nach THEN abgearbeitet. Ist die Bedingung nicht erfüllt, werden die Befehle nach ELSE abgearbeitet.

Syntax : **AGAIN...:ELSE...**
Parameter : keine
Funktion : War die Bedingung bei der letzten CASE-Anweisung wahr, so werden die Befehle zwischen AGAIN und ELSE abgearbeitet. Andernfalls die Befehle nach ELSE.

Syntax : **JUMP label**
Parameter : label — beliebige Zeichenkette
Funktion : Springt an die durch label gekennzeichnete Programmstelle.

Syntax : **LABEL label**
Parameter : wie »JUMP«
Funktion : Kennzeichnet einen Programmpunkt für den Aufruf durch JUMP oder CALL.
Hinweis : Trifft das Programm ohne JUMP- oder CALL-Aufruf auf ein LABEL, wird der nächste Befehl gesucht und abgearbeitet.

Syntax : **CALL label**
Parameter : wie »JUMP«
Funktion : Ruft ein Unterprogramm auf.

Syntax : **SUBEND**
Parameter : keine
Funktion : Das durch CALL aufgerufene Unterprogramm wird wieder verlassen.

Funktionen

Syntax : **XCORD (egal)**
Parameter : egal — ohne Bedeutung
Funktion : Holt die x-Koordinate des Cursors.

Syntax : **YCORD (egal)**
Parameter : wie »XCORD«
Funktion : Holt die y-Koordinate des Cursors.

Syntax : **CODE (x) , y**
Parameter : x — x-Koordinate (0 bis 39)
y — y-Koordinate(0 bis 24)
Funktion : Holt den Bildschirmcode des Zeichens an der x/y-Position.

Syntax : **FARB (x) , y**
Parameter : wie »CODE«
Funktion : Holt den Farbcode des Zeichens an der x/y-Position.

Syntax : **STICK (n)**
Parameter : n = 0 Port 2
= 1 Port 1
Funktion : Holt die Stellung des Joysticks (Tabelle 2).

Syntax : **CHECK (0)sn , sn**
Parameter : sn — Spritenummern (0 bis 7)
Funktion : Ergebnis:
= 1 die beiden Sprites kollidieren
= 0 die beiden Sprites kollidieren nicht

CHECK (1)sn
Parameter : sn — Spritenummer (0 bis 7)
Funktion : Ergebnis:
= 1 das Sprite kollidiert mit dem Hintergrund
= 0 das Sprite kollidiert nicht mit dem Hintergrund

Syntax : **TEST (egal)x , y**
Parameter : x — x-Koordinate
y — y-Koordinate
Funktion : Prüft, ob ein Punkt in der hochauflösenden Grafik gesetzt ist. Das Ergebnis entspricht dem Zeichenmodus.

Besonderheiten

Beim FILL/ZFILL/FFILL-Befehl muß die linke obere Ecke auch wirklich die linke obere Ecke des Rechtecks sein, da der Computer sonst unweigerlich »abstürzt«.

Der TRACE-Befehl verträgt sich leider nicht mit den Befehlen, die den Interrupt beeinflussen (VEC, OLDVEC, B/G/BG FLASH). Dies läßt sich aber einfach durch Drücken der RUN/STOP-RESTORE-Taste, nachdem TRACE eingegeben wurde, ändern. Die Unverträglichkeit des TRACE-Befehls und der »Interrupt-Befehle« resultiert aus der Umlenkung des Interrupts auf einen Rasterzeileninterrupt beim TRACE-Befehl. Dieser Rasterzeileninterrupt sorgt dafür, daß beim TRACE-Befehl die obersten drei Zeilen immer in Blockgrafik angezeigt werden.

Aber auch ohne TRACE-Modus kann man den Rasterzeileninterrupt nutzen. Er wird mit SYS 31851 aktiviert. Zuvor müssen Sie allerdings in die Speicherstellen 31889 und 31903 jeweils die Rasterzeile (0 bis 255), in der auf hochauflösende Grafik umgeschaltet wird, POKEn. Außerdem muß die Speicherstelle 31929 die Rasterzeile (0 bis 255) enthalten, in der wieder auf Textmodus zurückgeschaltet wird. Das Ausschalten dieses Rasterzeileninterrupts ist nur durch NRM oder RUN/STOP-RESTORE möglich.

Im TRACE-Modus funktioniert auch der GET-Befehl nicht mehr, da die Speicher der zuletzt gedrückten Taste ständig gelöscht werden.

Die Funktion TEST setzt wie alle anderen Grafikbefehle den Grafikcursor. Hiermit können Sie also vor jedem GPRINT den Grafikcursor an die gewünschte Position setzen.

Den Befehl @ sollten Sie nicht alleine verwenden. Der Computer könnte abstürzen. Das gleiche kann passieren, wenn Sie bei Befehlen einen oder mehrere Parameter vergessen.

Wenn ein HiRes-Bild auf Diskette gespeichert wird, blinkt die Floppy-LED, da das Programm hauptsächlich für Datasette geschrieben wurde. Das HiRes-Bild kann zwar später wieder geladen werden, jedoch nur in Schwarz-Weiß. Die Farb-Informationen sind verloren.

Fehlermeldungen

Ist ein Programm beendet oder tritt eine Fehlermeldung auf, wird der hochauflösende Grafikmodus ausgeschaltet. Bei einem Fehler wird die fehlerhafte Zeile gelistet. Bei den meisten Fehlern wird der ungefähre Fehlerpunkt negativ (invers) gekennzeichnet.

Zusätzliche Fehlermeldungen zum Basic V2:

UNTIL WITHOUT REPEAT
Tritt auf, wenn der Computer auf UNTIL trifft, ohne daß ein REPEAT vorausging.

LABEL NOT FOUND
Tritt auf, wenn ein Sprunglabel nicht vorhanden ist.

SUBEND WITHOUT CALL
Tritt auf, wenn der Computer auf SUBEND trifft, ohne daß ein CALL vorausging.

ILLEGAL SPRITE NUMBER
Tritt auf, wenn die Spritenummer sn größer als 7 ist.

ILLEGAL SPRITE STRING
Der String des DESIGN-Befehls enthält einen Fehler.

SPRITE DESIGNING COORDINATE
Tritt auf, wenn beim SPLOT-Befehl falsche x/y-Koordinaten angegeben werden. (Jens Schwarz/ah)

hex.-adr.	dez.-adr.	Funktion	hex.-adr.	dez.-adr.	Funktion
9EAA	40618	STICK	91B7	37303	YCORD
9E52	40530	CHECK	8EB1	36529	CODE
CCDC	52444	TEST	91C0	37312	FARB
8EAA	36522	XCORD			

Anfangsadressen der Funktionen

ASCII-Code	Tastenkombination	Funktion
1	CTRL + A	Schriftgröße in x-Richtung +1
2	CTRL + B	Schriftgröße in x-Richtung –1
3	CTRL + C	Schriftgröße in y-Richtung +1
4	CTRL + D	Schriftgröße in y-Richtung –1
22	CTRL + V	Cursor 1 Pixel nach links
16	CTRL + P	Cursor 1 Pixel nach rechts
15	CTRL + O	Cursor 1 Pixel nach oben
21	CTRL + U	Cursor 1 Pixel nach unten
157	SHIFT + CRSR lr	Cursor left
29	CRSR lr	Cursor right
145	SHIFT + CRSR ud	Cursor up
17	CRSR ud	Cursor down
19	HOME	Cursor home
147	CLR/HOME	siehe CLS
13	CTRL + M	Carriage Return
14	CTRL + N	Umschaltung Kleinbuchstaben
142	keine	Umschaltung Großbuchstaben
18	CTRL RVSON	RVS on
146	CTRL RVSOFF	RVS off
6	CTRL + F	Zeichenmodus 0
7	CTRL + G	Zeichenmodus 1
10	CTRL + J	Zeichenmodus 2
11	CTRL + K	Zeichenmodus 3
12	CTRL + L	Zeichenmodus 4

Tabelle 1. Steuerzeichen des GPRINT-Befehls

nichts	0
oben	1
unten	2
links	4
rechts	8
oben links	5
oben rechts	9
unten links	6
unten rechts	10
Feuer	Richtungszahl + 16

Tabelle 2. Ergebnisse der Joystick-Abfrage

dez.-adr.	hex.-adr.	Belegung
0— 1024	0000—0400	wie im Handbuch beschrieben
1025—28671	0401—6FFF	Basic-RAM
28672—32767	7000—7FFF	Editor- und Strukturbefehle
32768—33791	8000—83FF	Video-Ram (Bildschirm)
33792—34815	8400—87FF	Spriteblöcke 16 bis 31
34816—40959	8800—9FFF	Blockgrafik- und Spritegrafik-befehle
40960—49151	A000—BFFF	HRG
49152—53247	C000—CFFF	HRG-Befehle
53248—57343	D000—DFFF	keine Veränderungen
57344—59392	E000—E7FF	Speicher für Farben der HRG

Tabelle 3. Speicherbelegung

```
programm : s+g-basic    0801 4100
----------------------------------

0801 : 1a 08 c1 07 9e 32 30 39  1f
0809 : 36 20 20 20 48 53 53 20  08
0811 : 28 43 29 20 31 39 38 35  51
0819 : 00 00 00 ff ff 00 00 ff  19
0821 : ff 00 00 ff ff 00 00 ff  20
0829 : ff 00 00 ff ff 00 00 a9  7c
0831 : 00 85 57 85 59 a9 09 85  8c
0839 : 58 a9 88 85 5a a0 ff c8  75
0841 : b1 57 91 59 c8 d0 f9 e6  f6
0849 : 58 e6 5a a5 5a c9 a0 90  f7
0851 : ef a9 00 85 5b 85 5d a9  70
0859 : 21 85 58 a9 c0 85 5a a9  7d
0861 : 31 85 5c a9 70 85 5e a0  8f
0869 : ff c8 b1 57 91 59 b1 5b  85
0871 : 91 5d c8 d0 f5 e6 58 e6  c3
0879 : 5a e6 5c e6 5e a5 5e c9  5a
0881 : 80 90 e7 4c d0 9e 04 ff  df
0889 : ff 00 00 ff ff 00 00 ff  88
0891 : ff 00 00 ff ff 00 00 ff  90
0899 : ff 00 00 ff ff 00 00 ff  98
08a1 : ff 00 00 ff ff 00 00 ff  a0
08a9 : ff 00 00 ff ff 00 00 ff  a8
08b1 : ff 00 00 ff ff 00 00 ff  b0
08b9 : ff 00 00 ff ff 00 00 ff  b8
08c1 : ff 00 00 ff ff 00 00 ff  c0
08c9 : ff 00 00 ff ff 00 00 ff  c8
08d1 : ff 00 00 ff ff 00 00 ff  d0
08d9 : ff 00 00 ff ff 00 00 ff  d8
08e1 : ff 00 00 ff ff 00 00 ff  e0
08e9 : ff 00 00 ff ff 00 00 ff  e8
08f1 : ff 00 00 ff ff 00 00 ff  f0
08f9 : ff 00 00 ff ff 00 00 a6  46
0901 : 7a a0 04 84 0f bd 00 02  40
0909 : 10 07 c9 ff f0 3e e8 d0  55
0911 : f4 c9 20 f0 37 85 08 c9  63
0919 : 22 f0 56 24 0f 70 2d c9  8a
0921 : 3f d0 04 a9 99 d0 25 c9  47
0929 : 30 90 04 c9 3c 90 1d 84  a1
0931 : 71 a0 00 84 0b 88 86 7a  87
0939 : ca c8 e8 bd 00 02 38 f9  3e
0941 : 9e a0 f0 f5 c9 80 d0 30  6e
0949 : 05 0b a4 71 e8 c8 99 fb  5e
0951 : 01 b9 fb 01 f0 38 38 e9  d3
0959 : 3a f0 04 c9 49 d0 02 85  74
0961 : 0f 38 e9 55 d0 9f 85 08  e1
0969 : bd 00 02 f0 df c5 08 f0  f3
0971 : db c8 99 fb 01 e8 d0 f0  13
0979 : a6 7a e6 0b c8 b9 9d a0  89
0981 : 10 fa b9 9e a0 d0 b4 f0  96
0989 : 0f bd 00 02 10 bc 99 fd  00
0991 : 01 c6 7b a9 ff 85 7a 60  e0
0999 : a9 01 85 0b a0 00 b9 3c  ef
09a1 : 8a d0 02 c8 e8 bd 00 02  ad
09a9 : 38 f9 3c 8a f0 f5 c9 80  25
09b1 : d0 0e e6 71 a4 71 a9 cc  86
09b9 : 99 fb 01 a5 0b 4c 4b 88  96
09c1 : a6 7a e6 0b c8 b9 3b 8a  1c
09c9 : 10 fa b9 3c 8a d0 d6 a9  2a
09d1 : 01 85 0b a0 00 b9 3c 8b  42
09d9 : d0 02 c8 e8 bd 00 02 38  4e
09e1 : f9 3c 8b f0 f5 c9 80 d0  4a
09e9 : 0e e6 71 a4 71 a9 cd 99  2a
09f1 : fb 01 a5 0b 4c 4b 88 a6  c6
09f9 : 7a e6 0b c8 b9 3b 8b 10  86
0a01 : fa b9 3c 8b d0 [illegible] 4c 8a  62
0a09 : 88 10 13 24 0f 30 0f c9  25
0a11 : ff f0 0b c9 cc f0 0a c9  94
0a19 : cd f0 27 4c 24 a7 4c f3  4a
0a21 : a6 c8 84 49 b1 5f aa a0  78
0a29 : ff ca f0 08 c8 b9 3c 8a  2b
0a31 : 10 fa 30 f5 c8 b9 3c 8a  e9
0a39 : 30 05 20 47 ab d0 f5 4c  8e
0a41 : ef a6 c8 84 49 b1 5f aa  3b
0a49 : a0 ff ca f0 08 c8 b9 3c  e0
0a51 : 8b 10 fa 30 f5 c8 b9 3c  ae
0a59 : 8b 30 05 20 47 ab d0 f5  43
0a61 : 4c ef a6 20 73 00 20 6d  e5
0a69 : 89 4c ae a7 c9 cc f0 0a  94
0a71 : c9 cd f0 11 20 79 00 4c  e6
0a79 : ed a7 20 73 00 38 e9 01  1c
0a81 : 0a aa 4c 8e 89 20 73 00  2d
0a89 : 18 69 36 0a aa bd 29 8c  7b
0a91 : 48 bd 28 8c 48 4c 73 00  08
0a99 : a9 00 85 0d 20 73 00 c9  77
0aa1 : cd f0 06 20 79 00 4c 8d  50
0aa9 : ae 20 73 00 c9 30 90 f3  8c
0ab1 : 18 69 36 0a 48 20 73 00  a0
0ab9 : 20 f1 ae 68 a8 b9 28 8c  9d
0ac1 : 85 55 b9 29 8c 85 56 20  13
0ac9 : 54 00 4c 8d ad 48 8a 48  ba
0ad1 : 98 48 a9 7f 8d 0d dd ac  fa
0ad9 : 0d dd 10 03 4c 72 fe 20  ce
0ae1 : 02 fd d0 03 6c 02 80 20  8f
0ae9 : bc f6 20 e1 ff f0 03 4c  91
0af1 : 72 fe 20 15 fd 20 a3 fd  f8
0af9 : 20 18 e5 a9 05 8d 18 d0  93
0b01 : a9 01 8d 00 dd a9 80 8d  d6
0b09 : 88 02 a9 ce 8d 18 03 a9  cf
0b11 : 89 8d 19 03 a9 1b 8d 11  d3
0b19 : d0 a9 00 8d aa 02 8d ab  b8
0b21 : 02 8d ac 02 8d a1 02 8d  5e
0b29 : a2 02 8d a3 02 8d a4 02  c7
0b31 : a9 01 8d a5 02 20 44 e5  71
0b39 : 6c 02 a0 41 d4 53 45 d4  9d
0b41 : 46 53 45 d4 5a 53 45 d4  1c
0b49 : 43 4f 4c 4f 55 d2 43 55  d4
0b51 : 52 53 4f d2 46 49 4c cc  f4
0b59 : 49 4e d6 46 46 49 4c cc  c1
0b61 : 5a 46 49 4c cc 56 45 c3  d6
0b69 : 4f 4c 44 56 45 c3 42 46  c2
0b71 : 4c 41 53 c8 47 46 4c 41  a6
0b79 : 53 c8 42 47 46 4c 41 53  1c
0b81 : c8 46 4f 46 c6 44 4f 57  83
0b89 : ce 4c 45 46 d4 52 49 47  2b
0b91 : 48 d4 55 d0 46 4c 41 53  25
0b99 : c8 4f 46 c6 53 43 4c d2  99
0ba1 : 53 43 4f 4c 4f 55 d2 53  85
0ba9 : 52 56 d3 53 42 4c 4f 43  d0
0bb1 : cb 53 4f ce 53 4f 46 c6  2a
0bb9 : 53 4d 4f 44 c5 4d 43 4f  82
0bc1 : 4c 4f 55 d2 58 45 58 d0  17
0bc9 : 01 ff 59 45 58 d0 53 50  c3
0bd1 : 4c 4f d4 53 53 45 d4 53  be
0bd9 : 55 d0 53 44 4f 57 ce 53  85
0be1 : 52 49 47 48 d4 53 4c 45  56
0be9 : 46 d4 53 50 52 49 4f d2  cb
0bf1 : 42 4d 4f 56 c5 52 52 49  43
0bf9 : 47 48 d4 52 4c 45 46 d4  95
0c01 : 52 55 d0 52 44 4f 57 ce  36
0c09 : 53 4d 4f 56 c5 4d 4f 56  52
0c11 : 45 54 cf 44 45 53 49 47  9f
0c19 : ce 58 4d 49 d2 59 4d 49  50
0c21 : d2 53 43 4f 52 c4 54 55  9f
0c29 : 52 ce 52 4f 54 d2 52 4f  25
0c31 : 54 cc 44 45 54 45 43 d4  cb
0c39 : 00 00 ff 48 52 c7 4e 52  83
0c41 : cd 43 4c d3 4d 55 4c 54  97
0c49 : c9 4d 4f 44 c5 47 43 4f  57
0c51 : 4c 4f 55 d2 50 4c 4f d4  43
0c59 : 44 52 41 d7 4c 49 4e c5  e5
0c61 : 52 45 c3 42 4c 4f 43 cb  73
0c69 : 41 4e 47 cc 41 52 c3 47  81
0c71 : 50 52 49 4e d4 47 53 41  5e
0c79 : 56 c5 47 4c 4f 41 c4 46  ac
0c81 : 4b 45 d9 50 41 47 c5 4d  ef
0c89 : 4f 56 c5 42 41 43 cb 4f  b9
0c91 : 4c c4 44 49 53 d0 44 55  f1
0c99 : 4d d0 4d 45 52 47 c5 5a  76
0ca1 : 4c 4f 41 c4 5a 53 41 56  70
0ca9 : c5 40 48 45 4c d0 48 45  40
0cb1 : 4c d0 44 45 cc 4d 45 cd  07
0cb9 : 41 52 52 41 d9 c0 54 52  7a
0cc1 : 41 43 c5 52 45 54 52 41  22
0cc9 : 43 c5 46 49 4e c4 41 55  64
0cd1 : 54 cf 52 45 50 45 41 d4  28
0cd9 : 55 4e 54 49 cc 45 4c 53  62
0ce1 : c5 43 41 53 c5 41 47 41  09
0ce9 : 49 ce 4a 55 4d d0 4c 41  e6
0cf1 : 42 45 cc 43 4f 4c c4 45  66
0cf9 : 52 52 4f d2 43 41 4c cc  ac
0d01 : 53 55 42 45 4e c4 53 54  39
0d09 : 49 43 cb 43 48 45 43 cb  a2
0d11 : 54 45 53 d4 58 43 4f 52  f9
0d19 : c4 59 43 4f 52 c4 43 4f  3c
0d21 : 44 c5 46 41 52 c2 00 a7  8c
0d29 : 8d b4 8d cb 8d dd 8d ec  c5
0d31 : 8d ff 8d 28 8e 4e 8e 6a  91
0d39 : 8e 8b 8e f6 8e 03 8f 29  a1
0d41 : 8f 7b 8f 9a 8f 6e 8f bc  e9
0d49 : 8f 19 90 72 90 c5 90 20  91
0d51 : 91 a9 91 e5 91 18 92 f4  e6
```

Listing des Sprite+Grafik-Basic. Bitte mit dem MSE eingeben.

```
0d59 : 91 2a 92 85 92 91 92 9d   90
0d61 : 92 c2 92 d5 92 f3 92 f3   af
0d69 : 92 11 93 2d 94 62 94 82   c2
0d71 : 94 a2 94 e2 94 22 95 45   13
0d79 : 95 76 95 dc 95 42 96 a5   5b
0d81 : 96 db 97 6f 99 c2 99 f8   e1
0d89 : 9f 87 9c 32 97 14 9c 86   f3
0d91 : 9d 15 9e 40 9e ff bf 91   74
0d99 : c5 6f c0 f9 c0 4c c1 5a   af
0da1 : c1 23 c2 30 c5 58 c5 1c   19
0da9 : c6 c0 c6 9c c7 0f c8 0d   47
0db1 : cb 57 cd 0f ce 7a 71 63   ca
0db9 : 72 79 72 de 72 f5 72 25   4b
0dc1 : 73 d0 73 90 74 bf 74 dd   5e
0dc9 : 74 56 75 dc 75 65 76 d4   67
0dd1 : 76 33 7b d8 7c d8 7c e5   27
0dd9 : 7c fb 77 ad 78 fd 7c 1b   86
0de1 : 7d 3a a9 59 7d 9e 7d a4   1d
0de9 : 7d f7 a8 cf 9e f2 7e 5c   ba
0df1 : 7f 76 7f aa 9e 52 9e dc   91
0df9 : cc aa 8e b7 91 b1 8e c0   17
0e01 : 91 00 d8 d8 d8 d8 d8 d8   4d
0e09 : d8 d9 d9 d9 d9 d9 d9 da   09
0e11 : da da da da da da db db   16
0e19 : db db db db 02 00 00 01   76
0e21 : 01 f0 00 01 02 00 00 14   03
0e29 : 00 ee 00 00 0c 88 15 ff   fa
0e31 : ff ce ff 00 db bd f0 ec   e0
0e39 : 85 fb b5 d9 29 03 0d 88   54
0e41 : 02 85 fc 60 20 36 8d b1   9e
0e49 : fb 60 48 20 36 8d 68 91   1f
0e51 : fb 60 bd 03 8d 85 fc bd   c1
0e59 : f0 ec 85 fb 60 20 53 8d   10
0e61 : b1 fb 60 48 20 53 8d 68   d5
0e69 : 91 fb 60 48 20 36 8d 68   d4
0e71 : 91 fb bd 03 8d 85 fc a5   14
0e79 : fd 91 fb 60 20 fd ae 20   37
0e81 : 9e b7 60 c0 28 b0 05 e0   09
0e89 : 19 b0 01 60 a2 0e 4c 37   81
0e91 : a4 e0 10 b0 01 60 4c 8d   1f
0e99 : 8d 20 9e b7 8a 48 20 7d   3b
0ea1 : 8d 68 a8 20 84 8d 60 20   07
0ea9 : 9a 8d 18 20 f0 ff 20 fd   9f
0eb1 : ae 4c a0 aa 20 9e b7 8a   ee
0eb9 : 48 20 7d 8d 20 92 8d 86   fc
0ec1 : fd 20 fd ae 20 9a 8d 68   01
0ec9 : 4c 6c 8d 20 9e b7 20 92   00
0ed1 : 8d 8a 48 20 fd ae 20 9a   c4
0ed9 : 8d 68 4c 64 8d 20 9e b7   fe
0ee1 : 8a 48 20 fd ae 20 9a 8d   c9
0ee9 : 68 4c 4b 8d 20 9e b7 20   12
0ef1 : 92 8d 8e 20 d0 20 7d 8d   11
0ef9 : 20 92 8d 8e 21 d0 60 20   f2
0f01 : 9e b7 20 92 8d 8e 86 02   41
0f09 : 60 20 9e b7 86 57 20 7d   b6
0f11 : 8d 86 58 a4 57 20 84 8d   30
0f19 : 20 7d 8d 86 59 20 7d 8d   d4
0f21 : 86 5a a4 59 20 84 8d 60   46
0f29 : 20 0a 8e 20 7d 8d 86 5b   0b
0f31 : 20 7d 8d 20 92 8d 86 fd   23
0f39 : a4 57 88 c8 a6 58 ca e8   ee
0f41 : a5 5b 20 6c 8d e4 5a d0   34
0f49 : f6 c4 59 d0 ee 60 20 0a   98
0f51 : 8e a4 57 88 c8 a6 58 ca   d1
0f59 : e8 20 45 8d 49 80 20 4b   04
0f61 : 8d e4 5a d0 f3 c4 59 d0   7d
0f69 : eb 60 20 0a 8e 20 7d 8d   c9
0f71 : 20 92 8d 86 5b a4 57 88   58
0f79 : c8 a6 58 ca e8 a5 5b 20   6d
0f81 : 64 8d e4 5a d0 f6 c4 59   bb
0f89 : d0 ee 60 20 0a 8e 20 7d   7d
0f91 : 8d 86 5b a4 57 88 c8 a6   f7
0f99 : 58 ca e8 a5 5b 20 4b 8d   44
0fa1 : e4 5a d0 f6 c4 59 d0 ee   fd
0fa9 : 60 38 20 f0 ff 4c a2 b3   a0
0fb1 : 20 f7 b7 a5 14 48 20 7d   6e
0fb9 : 8d 68 a8 20 84 8d 20 45   68
0fc1 : 8d a8 4c a2 b3 ee 1d 8d   4c
0fc9 : ad 1d 8d c9 06 90 05 a9   ee
0fd1 : 00 8d 1d 8d c9 01 f0 0f   17
0fd9 : c9 02 f0 0e c9 03 f0 0d   34
0fe1 : c9 04 f0 0c 4c f4 8e 4c   a9
0fe9 : f4 8e 4c 0d 71 4c f4 8e   43
0ff1 : 4c f4 8e 4c 31 ea 78 a9   84
0ff9 : 8e 8d 15 03 a9 c6 8d 14   23
1001 : 03 58 60 78 a9 ea 8d 15   aa
1009 : 03 a9 31 8d 14 03 58 60   5a
1011 : 20 9e b7 20 92 8d 8e 1f   80
1019 : 8d 20 7d 8d 20 92 8d 8e   b1
1021 : 20 8d 20 7d 8d 8e 21 8d   ac
1029 : 60 20 11 8f 78 a9 3a 8d   a8
1031 : e9 8e a9 8f 8d ea 8e 58   d9
1039 : 60 ad 20 d0 8d 22 8d 20   f2
1041 : 49 8f 8d 20 d0 4c 31 ea   c3
1049 : ee 1e 8d ad 1e 8d cd 21   27
1051 : 8d f0 04 ad 22 8d 60 a9   70
1059 : 00 8d 1e 8d ad 22 8d 29   ce
1061 : 0f cd 1f 8d f0 04 ad 1f   f4
1069 : 8d 60 ad 20 8d 60 78 a9   a7
1071 : 8e 8d ea 8e a9 f4 8d e9   9e
1079 : 8e 58 60 20 11 8f 78 a9   12
1081 : 8c 8d e9 8e a9 8f 8d ea   43
1089 : 8e 58 60 ad 21 d0 8d 22   24
1091 : 8d 20 49 8f 8d 21 d0 4c   30
1099 : 31 ea 20 11 8f 78 a9 ab   24
10a1 : 8d e9 8e a9 8f 8d ea 8e   2a
10a9 : 58 60 ad 21 d0 8d 22 8d   de
10b1 : 20 49 8f 8d 21 d0 8d 20   1a
10b9 : d0 4c 31 ea 20 0a 8e a5   31
10c1 : 5a c9 18 d0 02 c6 5a a6   2d
10c9 : 5a e8 ca bd f0 ec 85 fb   86
10d1 : 85 5b b5 d9 29 03 0d 88   9c
10d9 : 02 85 fc bd 03 8d 85 5c   00
10e1 : bd f1 ec 85 fd 85 5d b5   6f
10e9 : da 29 03 0d 88 02 85 fe   67
10f1 : bd 04 8d 85 5e a4 57 88   3e
10f9 : c8 b1 fb 91 fd b1 5b 91   c9
1101 : 5d c4 59 d0 f3 e4 58 d0   9a
1109 : c1 a4 57 a6 58 88 c8 a9   07
1111 : 20 20 4b 8d c4 59 d0 f6   0e
1119 : 60 20 0a 8e a5 57 d0 02   3a
1121 : e6 57 a6 58 ca e8 b5 d9   e6
1129 : 29 03 0d 88 02 85 fc 85   74
1131 : fe bd 03 8d 85 5c 85 5e   8e
1139 : bd f0 ec 85 fb 85 5b c9   47
1141 : 00 d0 04 c6 fe c6 5e 38   93
1149 : e9 01 85 fd 85 5d a4 57   58
1151 : 88 c8 b1 fb 91 fd b1 5b   b0
1159 : 91 5d c4 59 d0 f3 e4 5a   ea
1161 : d0 c3 a4 59 a6 58 ca e8   91
1169 : a9 20 20 4b 8d e4 5a d0   9f
1171 : f6 60 20 0a 8e a5 59 c9   f0
1179 : 27 d0 02 c6 59 a6 58 ca   23
1181 : e8 bd f0 ec 85 fb 85 5b   26
1189 : 18 69 01 85 fd 85 5d b5   34
1191 : d9 29 03 0d 88 02 85 fc   0a
1199 : 85 fe bd 03 8d 85 5c 85   ef
11a1 : 5e a4 59 c8 88 b1 fb 91   ea
11a9 : fd b1 5b 91 5d c4 57 d0   83
11b1 : f3 e4 5a d0 cb a4 57 a6   53
11b9 : 58 ca e8 a9 20 20 4b 8d   31
11c1 : e4 5a d0 f6 60 20 0a 8e   32
11c9 : a5 58 d0 02 e6 58 a6 58   8b
11d1 : ca e8 bd f0 ec 85 fb 85   93
11d9 : 5b b5 d9 29 03 0d 88 02   69
11e1 : 85 fc bd 03 8d 85 5c bd   a6
11e9 : ef ec 85 fd 85 5d b5 d8   3b
11f1 : 29 03 0d 88 02 85 fe bd   b4
11f9 : 02 8d 85 5e a4 57 88 c8   a8
1201 : b1 fb 91 fd b1 5b 91 5d   cb
1209 : c4 59 d0 f3 e4 5a d0 c1   14
1211 : a4 57 a6 5a 88 c8 a9 20   0b
1219 : 20 4b 8d c4 59 d0 f6 60   93
1221 : 20 9e b7 20 92 8d 8e 23   98
1229 : 8d 20 7d 8d 20 92 8d 8e   c1
1231 : 24 8d 20 7d 8d 20 92 8d   13
1239 : 8e 25 8d 78 a9 49 8d e6   b5
1241 : 8e a9 91 8d e7 8e 58 60   cf
1249 : ee 26 8d ad 26 8d c9 03   5f
1251 : d0 05 a9 00 8d 26 8d a5   9a
1259 : fd 8d 27 8d a5 fe 8d 28   71
1261 : 8d a9 00 85 fd a9 db 85   1b
1269 : fe a0 e7 20 89 91 88 c0   7e
1271 : ff d0 f8 c6 fe a5 fe c9   9c
1279 : d7 d0 f0 ad 27 8d 85 fd   9b
1281 : ad 28 9d 85 fe 4c 31 ea   47
1289 : b1 fd 29 0f ae 26 8d dd   73
1291 : 23 8d f0 01 60 a5 fe 48   97
1299 : 29 03 0d 88 02 85 fe b1   44
12a1 : fd 49 80 91 fd 68 85 fe   cc
12a9 : 60 78 a9 8e 8d e7 8e a9   27
12b1 : f4 8d e6 8e 58 60 38 20   a1
12b9 : f0 ff 8a a8 4c a2 b3 20   49
12c1 : f7 b7 a5 14 48 20 7d 8d   16
12c9 : 68 a8 20 84 8d 20 5e 8d   8c
12d1 : 29 0f a8 4c a2 b3 20 8a   93
12d9 : ad 20 f7 b7 a5 14 85 fb   94
12e1 : a5 15 85 fc 60 20 d7 91   9b
12e9 : a0 ff a9 00 c8 91 fb c0   7e
12f1 : 3f d0 f9 60 20 d7 91 20   6a
12f9 : c4 9e a0 ff c8 a9 ff 38   7f
1301 : f1 fb 91 fb c0 3f d0 f4   07
1309 : 4c ca 9e 20 9e b7 e0 08   a1
1311 : 90 05 a2 21 4c 37 a4 60   c2
1319 : 20 0c 92 86 fd 20 7d 8d   a7
1321 : 20 92 8d 8a a6 fd 9d 27   5e
1329 : d0 60 20 0c 92 86 fd 20   49
1331 : 7d 8d ea ea ea 86 fe a9   bf
1339 : f8 85 fb ad 88 02 18 69   74
1341 : 03 85 fc a4 fd a5 fe 91   07
1349 : fb 60 a9 01 c0 00 f0 05   d9
1351 : 0a 88 4c 4d 92 60 85 fb   96
1359 : 86 fc 20 4b 92 11 fb 91   94
1361 : fb 60 85 fb 86 fc 20 4b   d4
1369 : 92 51 fb 91 fb 60 85 fb   a6
1371 : 86 fc 20 4b 92 85 fd b1   97
1379 : fb 25 fd 85 fd b1 fb 38   05
1381 : e5 fd 91 fb 60 20 0c 92   a5
1389 : 8a a8 a9 15 a2 d0 4c 57   05
1391 : 92 20 0c 92 8a a8 a9 15   47
1399 : a2 d0 4c 6f 92 20 0c 92   24
13a1 : 86 fd 20 7d 8d e0 02 b0   27
13a9 : 16 e0 00 f0 09 a4 fd a9   4e
13b1 : 1c a2 d0 4c 57 92 a4 fd   75
13b9 : a9 1c a2 d0 4c 6f 92 4c   56
13c1 : 28 b4 20 9e b7 20 92 8d   01
13c9 : 8e 25 d0 20 7d 8d 20 92   0c
13d1 : 8d 8e 26 d0 60 20 0c 92   a5
13d9 : 86 fd 20 7d 8d e0 02 b0   5f
13e1 : de a9 1d a4 fd e0 00 f0   38
13e9 : 05 a2 d0 4c 57 92 a2 d0   33
13f1 : 4c 6f 92 20 0c 92 86 fd   09
13f9 : 20 7d 8d e0 02 b0 c0 a9   53
1401 : 17 a4 fd e0 00 f0 05 a2   e7
1409 : d0 4c 57 92 a2 d0 4c 6f   e8
1411 : 92 20 8a ad 20 f7 b7 a5   f7
1419 : 14 85 57 a5 15 85 58 20   99
1421 : 7d 8d 86 fb 20 7d 8d 86   b7
1429 : fc e0 15 90 05 a2 23 4c   77
1431 : 37 a4 20 7d 8d e0 02 b0   bb
1439 : f4 e0 01 f0 58 a5 fc 0a   b6
1441 : 18 65 fc 85 fc a5 fb c9   7c
1449 : 18 b0 e2 4a 4a 4a 18 65   dd
1451 : fc 85 fc a5 fb 29 07 49   bb
1459 : 07 a8 20 4b 92 85 fd 20   b3
1461 : 7d 8d 20 c4 9e 20 6c 93   09
1469 : 4c ca 9e e0 03 90 01 60   57
1471 : e0 01 f0 0f e0 02 f0 14   fa
1479 : a4 fc b1 57 05 fd 45 fd   44
1481 : 91 57 60 a4 fc b1 57 05   2f
1489 : fd 91 57 60 a4 fc b1 57   d8
1491 : 45 fd 91 57 60 a5 fc 0a   5f
1499 : 18 65 fc 85 fc a5 fb c9   d4
14a1 : 0c b0 8a 4a 4a 18 65 fc   e6
14a9 : 85 fc a5 fb 0a 29 07 49   2e
14b1 : 07 85 fd 20 7d 8d 20 c4   4d
14b9 : 9e 20 c0 93 4c ca 9e e0   61
14c1 : 04 90 01 60 e0 01 f0 26   80
14c9 : e0 02 f0 37 e0 03 f0 45   42
14d1 : a4 fd 20 4b 92 85 fe c6   c4
14d9 : fd a4 fd 20 4b 92 18 65   20
14e1 : fe 85 fe a4 fc b1 57 05   bb
14e9 : fe 45 fe 91 57 60 a4 fd   83
14f1 : 20 4b 92 a4 fc 11 57 91   c9
14f9 : 57 c6 fd a4 fd 20 4b 92   fa
1501 : 4c e2 93 c6 fd a4 fd 20   b9
1509 : 4b 92 a4 fc 11 57 91 57   27
1511 : e6 fd 4c fc 93 a4 fd 20   3f
1519 : 4b 92 85 fe c6 fd a4 fd   d9
1521 : 20 4b 92 18 65 fe a4 fc   69
1529 : 11 57 91 57 60 20 0c 92   91
1531 : 86 fd 8a a8 20 4b 92 85   1f
1539 : fe 20 fd ae 20 eb b7 8a   f2
1541 : a8 a5 fd 0a aa 98 9d 01   64
1549 : d0 a5 14 9d 00 d0 a5 15   ec
1551 : a4 fd a2 d0 c9 00 d0 05   a0
1559 : a9 10 4c 6f 92 a9 10 4c   5b
1561 : 57 92 20 0c 92 8a 0a 85   3c
1569 : fe 20 7d 8d 86 fb a6 fe   69
1571 : bd 01 d0 38 e5 fb 90 04   72
1579 : 9d 01 d0 60 a9 00 9d 01   ea
1581 : d0 60 20 0c 92 8a 0a 85   bc
1589 : fe 20 7d 8d 86 fb a6 fe   89
1591 : bd 01 d0 18 65 fb b0 04   07
1599 : 9d 01 d0 60 a9 ff 9d 01   0a
15a1 : d0 60 20 0c 92 8a 0a 85   dc
15a9 : fe 20 7d 8d 86 fb a6 fe   a9
15b1 : bd 00 d0 18 65 fb b0 04   a6
15b9 : 9d 00 d0 60 9d 00 d0 a5   ff
15c1 : fe 4a a8 20 4b 92 85 fb   6a
15c9 : ad 10 d0 25 fb d0 0b a5   15
15d1 : fe 4a a8 a9 10 a2 d0 4c   46
15d9 : 57 92 a9 ff a6 fe 9d 00   bc
15e1 : d0 60 20 0c 92 8a 0a 85   1c
15e9 : fe 20 7d 8d 86 fb a6 fe   e9
15f1 : bd 00 d0 38 e5 fb 90 04   72
15f9 : 9d 00 d0 60 9d 00 d0 a5   3f
1601 : fe 4a a8 20 4b 92 85 fb   aa
1609 : ad 10 d0 25 fb f0 0b a5   56
1611 : fe 4a a8 a9 10 a2 d0 4c   86
1619 : 6f 92 a9 00 a6 fe 9d 00   15
1621 : d0 60 20 0c 92 86 fe 20   45
1629 : 7d 8d e0 02 f0 fc a9 1b   b9
1631 : a4 fe e0 00 f0 05 a2 d0   f0
1639 : 4c 57 92 a2 d0 4c 6f 92   7c
1641 : a2 d0 4c 6f 92 20 8a ad   fc
1649 : 20 f7 b7 a5 14 85 fb a5   b0
1651 : 15 85 fc 20 fd ae 20 8a   57
1659 : ad 20 f7 b7 a5 14 85 fd   18
1661 : a5 15 85 fe 20 c4 9e a0   b6
1669 : 00 88 c8 b1 fb 91 fd c0   db
1671 : 3f d0 f7 4c ca 9e 20 8a   d7
1679 : ad 20 f7 b7 a5 14 85 fb   34
1681 : a5 15 85 fc 20 c4 9e a0   96
1689 : fd c8 c8 c8 b1 fb 29 01   d7
1691 : 8d 29 8d b1 fb 4a 91 fb   9d
1699 : c8 b1 fb 29 01 8d 2a 8d   9e
16a1 : b1 fb 4a 91 fb c8 b1 fb   d9
16a9 : 29 01 8d 2b 8d b1 fb 4a   06
16b1 : 91 fb ad 2a 8d f0 06 b1   cc
16b9 : fb 09 80 91 fb 88 ad 29   98
16c1 : 8d f0 06 b1 fb 09 80 91   ab
16c9 : fb 88 ad 2b 8d f0 06 b1   b5
```

Listing des Sprite+Grafik-Basic. Bitte mit dem MSE eingeben.

```
16d1 : fb 09 80 91 fb c0 3c 90   7b
16d9 : b0 4c ca 9e 20 8a ad 20   83
16e1 : f7 b7 a5 14 85 fb a5 15   99
16e9 : 85 fc 20 c4 9e a0 fd c8   05
16f1 : c8 c8 b1 fb 29 80 8d 29   28
16f9 : 8d b1 fb 0a 91 fb c8 b1   1e
1701 : fb 29 80 8d 2a 8d b1 fb   30
1709 : 0a 91 fb c8 b1 fb 29 80   94
1711 : 8d 2b 8d b1 fb 0a 91 fb   1c
1719 : ad 29 8d f0 06 b1 fb 09   cc
1721 : 01 91 fb 88 ad 2b 8d f0   47
1729 : 06 b1 fb 09 01 91 fb 88   c5
1731 : ad 2a 8d f0 06 b1 fb 09   65
1739 : 01 91 fb c0 3c 90 b0 4c   bd
1741 : ca 9e 20 8a ad 20 f7 b7   df
1749 : a5 14 85 fb a5 15 85 fc   ec
1751 : 20 c4 9e a0 00 b1 fb 8d   28
1759 : 29 8d c8 b1 fb 8d 2a 8d   a1
1761 : c8 b1 fb 8d 2b 8d a0 00   54
1769 : c8 c8 c8 b1 fb 88 88 88   35
1771 : 91 fb c8 c8 c8 c8 b1 fb   dc
1779 : 88 88 88 91 fb c8 c8 c8   54
1781 : c8 b1 fb 88 88 88 91 fb   3d
1789 : c8 c8 c8 c0 3e d0 db a0   1b
1791 : 3c ad 29 8d 91 fb c8 ad   17
1799 : 2a 8d 91 fb c8 ad 2b 8d   2f
17a1 : 91 fb 4c ca 9e 20 8a ad   0d
17a9 : 20 f7 b7 a5 14 85 fb a5   10
17b1 : 15 85 fc 20 c4 9e a0 3e   0c
17b9 : b1 fb 8d 29 8d 88 b1 fb   cc
17c1 : 8d 2a 8d 88 b1 fb 8d 2b   5f
17c9 : 8d a0 3e 88 88 88 b1 fb   d2
17d1 : c8 c8 c8 91 fb 88 88 88   99
17d9 : 88 b1 fb c8 c8 c8 91 fb   63
17e1 : 88 88 88 88 b1 fb c8 c8   90
17e9 : c8 91 fb 88 88 88 c0 00   5a
17f1 : d0 db a0 02 ad 29 8d 91   95
17f9 : fb 88 ad 2a 8d 91 fb 88   4f
1801 : ad 2b 8d 91 fb 4c ca 9e   64
1809 : a0 ff 20 a2 b3 20 49 b8   d8
1811 : 20 49 b8 20 0c bc a0 ff   31
1819 : 20 a2 b3 a5 48 20 6a b8   cc
1821 : a9 41 85 45 a9 00 85 46   b2
1829 : 20 e7 b0 85 49 84 4a 4c   94
1831 : d0 bb 20 0c 92 8a 0a 85   19
1839 : fb 4a a8 20 4b 92 85 fc   e1
1841 : a6 fb bd 00 d0 a8 20 a2   6c
1849 : b3 ad 10 d0 25 fc f0 18   1f
1851 : a6 57 a4 00 20 d4 bb a0   a5
1859 : ff 20 a2 b3 a5 57 a4 00   2f
1861 : 20 8c ba a5 48 20 6a b8   cb
1869 : 20 fd ae 20 8b b0 aa 20   61
1871 : d4 bb a6 fb bd 01 d0 a8   c4
1879 : 20 a2 b3 20 fd ae 20 8b   c8
1881 : b0 aa 20 d4 bb 60 20 a2   ad
1889 : b3 a2 57 a0 00 20 d4 bb   43
1891 : a0 40 20 a2 b3 a9 57 a0   d5
1899 : 00 20 8c ba a5 48 20 2b   97
18a1 : ba 60 20 f7 b7 a5 15 f0   71
18a9 : 05 a2 0e 4c 37 a4 a5 14   64
18b1 : aa a9 f8 85 fd ad 88 02   92
18b9 : 18 69 03 85 fe 8a a8 b1   42
18c1 : fd a8 4c 87 97 85 fb 84   b5
18c9 : fc 20 4b 92 85 fd 25 fb   cf
18d1 : f0 09 a5 fe 05 fd 85 fe   e3
18d9 : 4c ff ff 20 0c 92 86 5c   51
18e1 : 8a a8 20 4b 92 85 5d 20   3c
18e9 : fd ae 20 8a ad a2 e0 a0   4b
18f1 : 9f 20 d4 bb 20 f7 b7 a5   39
18f9 : 15 c9 02 90 05 a2 0e 4c   bc
1901 : 37 a4 20 fd ae 20 8a ad   c4
1909 : a2 e5 a0 9f 20 d4 bb 20   92
1911 : f7 b7 a5 15 d0 e7 20 fd   b9
1919 : ae 20 8a ad a2 ea a0 9f   73
1921 : 20 d4 bb 20 f7 b7 a5 15   9c
1929 : c9 02 b0 d1 8d 2b 8e a5   11
1931 : 14 8d 2a 8e 20 fd ae 20   55
1939 : 8a ad a2 ef a0 9f 20 d4   72
1941 : bb 20 f7 b7 a5 15 d0 75   32
1949 : a5 14 8d 2c 8e 20 fd ae   20
1951 : 20 8a ad a2 f4 a0 9f 20   89
1959 : d4 bb 20 f7 b7 a5 15 c9   a2
1961 : 02 b0 7f a9 e0 a0 9f 20   a2
1969 : a2 bb a9 ea a0 9f 20 50   d9
1971 : b8 a2 ea a0 9f 20 d4 bb   0f
1979 : a9 e5 a0 9f 20 a2 bb a9   8a
1981 : ef a0 9f 20 50 b8 a2 ef   e1
1989 : a0 9f 20 d4 bb a9 f4 a0   ba
1991 : 9f 20 a2 bb a9 ea a0 9f   14
1999 : 20 0f bb a2 ea a0 9f 20   f6
19a1 : d4 bb a9 f4 a0 9f 20 a2   28
19a9 : bb a9 ef a0 9f 20 0f bb   f7
19b1 : a2 ef a0 9f 20 d4 bb a9   52
19b9 : f4 a0 9f 20 a2 bb 20 f7   62
19c1 : b7 a5 14 8d 33 8d ce 33   43
19c9 : 8d a5 15 8d 34 8d 20 fd   4c
19d1 : ae 20 9e b7 8e 56 99 a9   83
19d9 : e5 a0 9f 20 a2 bb 20 f7   73
19e1 : b7 a6 14 86 fb a9 e0 a0   93
19e9 : 9f 20 a2 bb 20 f7 b7 a6   a6
19f1 : fb a5 5c 85 fd 20 40 94   92
19f9 : a9 e5 a0 9f 20 a2 bb a9   0a
1a01 : ef a0 9f 20 67 b8 46 66   4e
1a09 : a2 e5 a0 9f 20 d4 bb 20   92
1a11 : f7 b7 a6 14 86 fb a9 e0   c1
1a19 : a0 9f 20 a2 bb a9 ea a0   db
1a21 : 9f 20 67 b8 46 66 a2 e0   a5
1a29 : a0 9f 20 d4 bb 20 f7 b7   47
1a31 : a5 5c 85 fd a6 fb 20 40   71
1a39 : 94 ad 33 8d 38 e9 01 8d   14
1a41 : 33 8d b0 08 ad 34 8d f0   fd
1a49 : 12 ce 34 8d a2 00 a0 00   2e
1a51 : c8 d0 fd e8 e0 01 d0 f8   69
1a59 : 4c f9 98 ad 2a 8e 85 14   d3
1a61 : ad 2b 8e 85 15 ae 2c 8e   8d
1a69 : a5 5c 85 fd 4c 40 94 20   b7
1a71 : 0c 92 86 5c 8a 0a 85 fb   fa
1a79 : 4a a8 20 4b 92 85 fc a6   1f
1a81 : fb bd 00 d0 a8 20 a2 b3   f2
1a89 : ad 10 d0 25 fc f0 19 a6   20
1a91 : 57 a4 00 20 d4 bb a0 ff   ec
1a99 : 20 a2 b3 a5 57 a4 00 20   87
1aa1 : 67 b8 20 49 b8 20 49 b8   b9
1aa9 : a2 e0 a0 9f 20 d4 bb a6   bc
1ab1 : fb bd 01 d0 a8 20 a2 b3   63
1ab9 : a2 e5 a0 9f 20 d4 bb 4c   9a
1ac1 : 17 98 20 8a ad 20 f7 b7   a9
1ac9 : a5 14 85 fb a5 15 85 fc   6c
1ad1 : 20 7d 8d e0 15 b0 09 86   37
1ad9 : fd 20 7d 8d e0 02 90 03   5e
1ae1 : 4c 48 b2 e0 00 f0 03 4c   46
1ae9 : 88 9a 20 fd ae 20 9e ad   48
1af1 : 20 a3 b6 c9 18 f0 05 a2   2c
1af9 : 22 4c 95 9a 20 c4 9e a0   de
1b01 : 18 88 b1 22 c9 41 f0 0c   90
1b09 : c9 49 f0 08 c9 42 f0 04   2e
1b11 : c9 2e d0 e3 c0 00 d0 e9   c5
1b19 : a5 fd 0a 18 65 fd 18 65   b4
1b21 : fb 85 fb a2 00 86 57 86   d1
1b29 : 58 a9 07 38 e5 58 a8 20   23
1b31 : 4b 92 85 fe a5 57 0a 0a   58
1b39 : 0a 18 65 58 a8 b1 22 c9   e8
1b41 : 41 d0 1f a4 57 b1 fb 05   44
1b49 : fe 91 fb e6 58 a4 58 c0   79
1b51 : 08 d0 d6 a0 00 84 58 e6   de
1b59 : 57 a4 57 c0 03 d0 ca 4c   6b
1b61 : ca 9e c9 42 f0 1e c9 49   ef
1b69 : f0 12 a4 57 b1 fb 25 fe   04
1b71 : 85 fe b1 fb 38 e5 fe 91   33
1b79 : fb 4c 4c 9a a4 57 b1 fb   c4
1b81 : 45 fe 91 fb 4c 4c 9a 20   fb
1b89 : fd ae 20 9e ad 20 a3 b6   91
1b91 : c9 0c f0 08 20 ca 9e a2   b6
1b99 : 22 4c 37 a4 20 c4 9e a0   28
1ba1 : 0c 88 b1 22 c9 [illegible] f0 14   35
1ba9 : c9 49 f0 10 c9 42 f0 0c   df
1bb1 : c9 43 f0 08 c9 44 f0 04   e3
1bb9 : c9 2e d0 d8 c0 00 d0 e1   fc
1bc1 : a5 fd 0a 18 65 fd 18 65   5c
1bc9 : fb 85 fb a9 00 85 57 85   4f
1bd1 : 58 a5 58 0a 85 5b a9 07   3b
1bd9 : 38 e5 5b a8 20 4b 92 85   a1
1be1 : 59 a9 06 38 e5 5b a8 20   b4
1be9 : 4b 92 85 5a a5 57 0a 0a   7b
1bf1 : 18 65 58 a8 b1 22 c9 41   bd
1bf9 : d0 2a a4 57 b1 fb 05 59   b4
1c01 : 91 fb 25 5a 85 5a b1 fb   0e
1c09 : 38 e5 5a 91 fb e6 58 a4   9e
1c11 : 58 c0 04 90 bc a9 00 85   01
1c19 : 58 e6 57 a4 57 c0 03 90   f7
1c21 : b0 4c ca 9e c9 44 f0 e5   cc
1c29 : c9 42 f0 25 c9 43 f0 37   dd
1c31 : c9 49 f0 40 a4 57 b1 fb   a7
1c39 : 25 59 85 59 b1 fb 38 e5   3f
1c41 : 59 91 fb 25 5a 85 5a b1   a5
1c49 : fb 38 e5 5a 91 fb 4c 0e   6b
1c51 : 9b a4 57 b1 fb 25 59 85   a4
1c59 : 59 b1 fb 38 e5 59 91 fb   f8
1c61 : 05 5a 91 fb 4c 0e 9b a4   64
1c69 : 57 b1 fb 05 59 05 5a 91   83
1c71 : fb 4c 0e 9b a4 57 b1 fb   4d
1c79 : 45 59 45 5a 91 fb 4c 0e   4e
1c81 : 9b 85 fb a9 00 85 fc a9   86
1c89 : 80 85 fd a9 01 85 fe a5   04
1c91 : fb 25 fd f0 06 a5 fc 05   48
1c99 : fe 85 fc 06 fe 46 fd d0   15
1ca1 : ee a5 fc 60 20 9e b7 e0   45
1ca9 : 02 90 05 a2 0e 4c 37 a4   f2
1cb1 : 86 5c 20 fd ae 20 8a ad   9f
1cb9 : 20 f7 b7 20 c4 9e a5 14   c7
1cc1 : 85 57 a5 15 85 58 a0 01   9d
1cc9 : b1 57 a6 5c f0 06 20 92   40
1cd1 : 9c 4c d8 9b 20 82 9b 91   e4
1cd9 : 57 c8 c8 c8 c0 3f 90 e8   fa
1ce1 : a0 00 b1 57 85 59 c8 c8   b0
1ce9 : b1 57 a6 5c f0 06 20 92   60
1cf1 : 9c 4c f8 9b 20 82 9b 88   fa
1cf9 : 88 91 57 c8 c8 a5 59 a6   a5
1d01 : 5c f0 06 20 92 9c 4c 0d   b4
1d09 : 9c 20 82 9b 91 57 c8 c0   42
1d11 : 3f 90 cf 60 20 9e b7 e0   30
1d19 : 02 90 05 a2 0e 4c 37 a4   62
1d21 : 86 5c 20 fd ae 20 8a ad   0f
1d29 : 20 f7 b7 20 c4 9e a5 14   37
1d31 : 85 57 a5 15 85 58 a0 00   0b
1d39 : b1 57 a6 5c f0 06 20 92   b0
1d41 : 9c 4c 48 9c 20 82 9b 85   39
1d49 : 59 84 5a a9 3e 38 e5 5a   a2
1d51 : a8 b1 57 a6 5c f0 06 20   22
1d59 : 92 9c 4c 61 9c 20 82 9b   85
1d61 : 85 5b a5 59 91 57 a4 5a   43
1d69 : a5 5b 91 57 c8 c0 1f 90   3b
1d71 : c7 a0 1f b1 57 a6 5c f0   84
1d79 : 06 20 92 9c 4c 83 9c 20   5b
1d81 : 82 9b 91 57 4c ca 9e a2   fb
1d89 : 00 86 5c 20 b6 9b 4c 37   cf
1d91 : 9c 85 fb a9 00 85 fc a9   97
1d99 : 80 85 5d a9 40 85 5e a9   65
1da1 : 02 85 fd a9 01 85 fe a5   9e
1da9 : fb 25 5d f0 06 a5 fc 05   38
1db1 : fd 85 fc a5 fb 25 5e f0   a9
1db9 : 06 a5 fc 05 fe 85 fc 06   8e
1dc1 : fd 06 fd 06 fe 06 fe 46   aa
1dc9 : 5d 46 5d 46 5e 46 5e d0   9d
1dd1 : d6 a5 fc 60 86 fd 84 fe   2d
1dd9 : 98 0a 18 65 fe 85 fe a5   8c
1de1 : fd 4a 4a 4a 18 65 fe 85   93
1de9 : fe a5 fd 29 07 85 fd a9   46
1df1 : 07 38 e5 fd a8 20 4b 92   2b
1df9 : 85 fd 60 20 d5 9c a4 fe   6c
1e01 : b1 fb 25 fd 60 85 58 20   8d
1e09 : d5 9c a5 58 f0 09 a4 fe   89
1e11 : b1 fb 05 fd 91 fb 60 a4   85
1e19 : fe b1 fb 05 fd 45 fd 91   b4
1e21 : fb 60 86 59 84 5a a9 14   03
1e29 : 38 e5 5a 85 5b a5 59 85   ee
1e31 : 5c a9 14 38 e5 59 85 5d   68
1e39 : a9 14 38 e5 5a 85 5e a5   4e
1e41 : 5a 85 5f a9 14 38 e5 59   b8
1e49 : 85 60 a6 59 a4 5a 20 fc   6b
1e51 : 9c 85 48 a6 5f a4 60 20   74
1e59 : fc 9c a6 59 a4 5a 20 06   22
1e61 : 9d a6 5d a4 5e 20 fc 9c   51
1e69 : a6 5f a4 60 20 06 9d a6   ea
1e71 : 5b a4 5c 20 fc 9c a6 5d   43
1e79 : a4 5e 20 06 9d a5 48 a6   8b
1e81 : 5b a4 5c 4c 06 9d 20 8a   b2
1e89 : ad 20 f7 b7 20 c4 9e a5   29
1e91 : 14 85 fb a5 15 85 fc a2   d2
1e99 : 0a a0 09 8a 48 98 48 20   32
1ea1 : 23 9d 68 a8 68 aa ca 10   e9
1ea9 : f2 a2 0a 88 10 ed 4c ca   b7
1eb1 : 9e 86 59 84 5a a9 14 38   2d
1eb9 : e5 5a 85 5b a5 59 85 5c   8c
1ec1 : a9 14 38 e5 59 85 5d a9   ca
1ec9 : 14 38 e5 5a 85 5e a5 5a   55
1ed1 : 85 5f a9 14 38 e5 59 85   16
1ed9 : 60 a6 59 a4 5a 20 fc 9c   4b
1ee1 : 85 48 a6 5b a4 5c 20 fc   47
1ee9 : 9c a6 59 a4 5a 20 06 9d   bd
1ef1 : a6 5d a4 5e 20 fc 9c a6   e4
1ef9 : 5b a4 5c 20 06 9d a6 5f   68
1f01 : a4 60 20 fc 9c a6 5d a4   3b
1f09 : 5e 20 06 9d a5 48 a6 5f   a3
1f11 : a4 60 4c 06 9d 20 8a ad   1a
1f19 : 20 f7 b7 20 c4 9e a5 14   27
1f21 : 85 fb a5 15 85 fc a2 0a   8f
1f29 : a0 09 8a 48 98 48 20 b2   ab
1f31 : 9d 68 a8 68 aa ca 10 f2   61
1f39 : a2 0a 88 10 ed 4c ca 9e   ae
1f41 : 20 9e b7 e0 02 90 03 4c   04
1f49 : 48 b2 aa a9 00 9d 1e d0   d1
1f51 : 60 20 f7 b7 a5 14 c9 02   dc
1f59 : b0 ed c9 01 f0 30 20 0c   bb
1f61 : 92 8a a8 20 4b 92 85 fb   be
1f69 : 20 fd ae 20 0c 92 8a a8   08
1f71 : 20 4b 92 85 fc ad 1e d0   e3
1f79 : 85 fd 25 fb f0 0b a5 fd   c0
1f81 : 25 fc f0 05 a0 01 4c a2   8a
1f89 : b3 a0 00 4c a2 b3 20 0c   76
1f91 : 92 8a a8 20 4b 92 85 fb   ee
1f99 : ad 1f d0 25 fb f0 05 a0   4b
1fa1 : 01 4c a2 b3 a0 00 4c a2   68
1fa9 : b3 20 f7 b7 a5 14 c9 02   87
1fb1 : 90 05 a2 0e 4c 37 a4 a9   93
1fb9 : ff 38 fd 00 dc 29 1f a8   39
1fc1 : 4c a2 b3 78 a9 35 85 01   b7
1fc9 : 60 a9 37 85 01 58 60 78   c2
1fd1 : a9 01 8d 00 dd a9 05 8d   b9
1fd9 : 18 d0 a9 80 8d 88 02 a9   4c
1fe1 : 00 8d 04 03 a9 88 8d 05   28
1fe9 : 03 a9 0a 8d 06 03 a9 89   27
1ff1 : 8d 07 03 a9 64 8d 08 03   d1
1ff9 : a9 89 8d 09 03 a9 99 8d   ea
2001 : 0a 03 a9 89 8d 0b 03 a9   b9
2009 : ce 8d 18 03 a9 89 8d 19   54
2011 : 03 a9 ac 8d 00 03 a9 7f   83
2019 : 8d 01 03 20 44 e5 a9 0b   1c
2021 : 8d 20 d0 8d 21 d0 a9 0a   f7
2029 : 8d 86 02 a9 0d 20 d2 ff   cc
2031 : a0 00 b9 78 9f 20 d2 ff   95
2039 : c8 c9 00 d0 f5 a9 00 85   b8
2041 : 33 85 37 a9 70 85 34 85   49
2049 : 38 a9 01 85 2b a9 04 85   62
2051 : 2c a9 00 8d 00 04 a9 00   ca
2059 : a8 91 2b c8 91 2b a5 2b   0d
2061 : 18 69 02 85 2d a5 2c 69   e3
2069 : 00 85 2e 20 8e a6 20 63   21
2071 : a6 58 a9 00 6c 00 03 20   c1
```

```
2079 : 20 20 20 20 20 20 20 20  79
2081 : 0e d3 50 52 49 54 45 20  64
2089 : 26 20 c7 52 41 46 49 4b  fd
2091 : 20 c2 41 53 49 43 0d 0d  ca
2099 : 20 20 20 20 20 20 20 20  99
20a1 : 20 20 20 20 46 4f 52 20  46
20a9 : c3 4f 4d 4d 4f 44 4f 52  0a
20b1 : 45 20 36 34 0d 0d 20 20  14
20b9 : 20 20 20 20 20 20 d7 52  fd
20c1 : 49 54 54 45 4e 20 42 59  94
20c9 : 20 ca 45 4e 53 20 d3 43  76
20d1 : 48 57 41 52 5a 0d 0d 00  a2
20d9 : 00 00 00 00 00 00 00 87  e9
20e1 : 5a 00 00 00 87 5a 00 00  87
20e9 : 00 88 1e 66 66 66 87 46  c6
20f1 : 00 00 00 84 20 00 00 00  84
20f9 : 20 a5 9b 4c ca 9e 00 a9  51
2101 : 00 8d a2 02 8d a3 02 8d  ca
2109 : a4 02 8d aa 02 8d ab 02  a6
2111 : 8d ac 02 a9 01 8d a5 02  c1
2119 : 8d a1 02 20 9e b7 20 92  49
2121 : 8d 8e a6 02 20 7d 8d 20  44
2129 : 92 8d 8e a9 02 a9 01 8d  e7
2131 : 00 dd a9 08 8d 18 d0 a9  bb
2139 : 3b 8d 11 d0 a9 c8 8d 16  dc
2141 : ff a9 00 85 fb 85 fd a9  fd
2149 : 83 85 fc a9 e3 85 fe 78  5a
2151 : a9 35 85 01 a0 f7 b1 fd  a3
2159 : 91 fb 88 c0 ff d0 f7 c6  16
2161 : fe c6 fc a5 fc c9 80 b0  38
2169 : ed a9 37 85 01 58 60 a9  51
2171 : 00 85 fb a9 bf 85 fc a0  c5
2179 : ff a9 00 91 fb 88 c0 ff  86
2181 : d0 f9 c6 fc a5 fc c9 a0  4a
2189 : b0 ef a9 00 85 fb 85 fd  e5
2191 : a9 83 85 fc a9 e3 85 fe  ca
2199 : a0 f7 ad a2 02 f0 11 ad  3c
21a1 : a6 02 0a 0a 0a 0a 0d a7  81
21a9 : 02 91 fb 91 fd 4c bf c0  68
21b1 : ad a6 02 0a 0a 0a 0a 0d  a6
21b9 : a9 02 91 fb 91 fd 88 c0  f4
21c1 : ff d0 d7 c6 fc c6 fe a5  44
21c9 : fc c9 80 b0 cd ad a2 02  b9
21d1 : f0 26 a9 00 85 fb 85 fd  89
21d9 : a9 e7 85 fe a9 db 85 fc  40
21e1 : a0 ff ad a8 02 91 fb 91  c1
21e9 : fd 88 c0 ff d0 f4 c6 fc  24
21f1 : c6 fe a5 fc c9 d8 b0 ea  3b
21f9 : 60 a9 01 8d a2 02 a9 d8  b2
2201 : 8d 16 d0 20 9e b7 20 92  1f
2209 : 8d 8e a6 02 20 7d 8d 20  2c
2211 : 92 8d 8e a7 02 20 7d 8d  35
2219 : 20 92 8d 8e a8 02 a9 00  f9
2221 : 85 fb 85 fd a9 db 85 fc  4e
2229 : a9 e7 85 fe 78 a9 35 85  bc
2231 : 01 a0 f7 b1 fd 91 fb 88  24
2239 : c0 ff d0 f7 c6 fe c6 fc  a5
2241 : a5 fc c9 d8 b0 ed a9 37  81
2249 : 85 01 58 60 20 9e b7 e0  09
2251 : 05 90 03 4c 48 b2 8e a5  88
2259 : 02 60 20 9e b7 20 92 8d  49
2261 : 8e a6 02 ad a2 02 d0 0a  0a
2269 : 20 7d 8d 20 92 8d 8e a9  d2
2271 : 02 60 20 7d 8d 20 92 8d  9a
2279 : 8e a7 02 20 7d 8d 20 92  49
2281 : 8d 8e a8 02 60 ac ac 02  e2
2289 : 98 29 f8 85 fe 85 fc a9  08
2291 : 00 85 fd 06 fc 26 fd 06  99
2299 : fc 26 fd 18 a5 fc 65 fe  01
22a1 : 85 fc a5 fd 69 00 85 fd  76
22a9 : 06 fc 26 fd 06 fc 26 fd  53
22b1 : 06 fc 26 fd 98 29 07 18  9e
22b9 : 65 fc 85 fc a5 fd 69 00  8d
22c1 : 85 fd 18 ad aa 02 29 f8  52
22c9 : 65 fc 85 fc ad ab 02 65  b8
22d1 : fd 85 fd 18 a9 a0 65 fd  44
22d9 : 85 fd ad aa 02 29 07 49  36
22e1 : 07 60 20 eb b7 a5 14 8d  b2
22e9 : aa 02 a5 15 8d ab 02 8e  fc
22f1 : ac 02 60 ad a2 02 d0 1b  20
22f9 : ad ab 02 f0 0b c9 02 b0  83
2301 : 11 ad aa 02 c9 40 b0 0a  49
2309 : ad ac 02 c9 c8 b0 03 4c  7d
2311 : 2a c2 60 ad ab 02 d0 0a  8c
2319 : ad aa 02 c9 a0 b0 03 4c  09
2321 : 09 c2 60 20 e3 c1 4c f4  0f
2329 : c1 20 86 c1 a8 ad a2 02  5b
2331 : d0 7d 20 4b 92 ae a5 02  6b
2339 : f0 20 e0 01 f0 0e 78 a2  38
2341 : 35 86 01 a0 00 51 fc 91  af
2349 : fc 4c 6a c2 78 a2 35 86  dd
2351 : 01 a0 00 11 fc 91 fc 4c  ad
2359 : 6a c2 78 a2 35 86 01 a0  64
2361 : 00 85 02 11 fc 45 02 91  ec
2369 : fc a2 37 86 01 58 ad ab  36
2371 : 02 f0 0a ad aa 02 4a 18  38
2379 : 69 80 4c 82 c2 ad aa 02  ce
2381 : 4a 4a 4a a8 ad ac 02 4a  75
2389 : 4a 4a aa b5 d9 29 03 09  5f
2391 : e0 85 fc 29 03 09 80 85  1e
2399 : fe bd f0 ec 85 fb 85 fd  9a
23a1 : ad a6 02 0a 0a 0a 0a 0d  96
23a9 : a9 02 91 fb 91 fd 60 ad  1d
23b1 : aa 02 85 59 ad ab 02 85  34
23b9 : 5a 18 0e aa 02 90 05 a9  04
23c1 : 01 8d ab 02 20 86 c1 85  fc
23c9 : 02 a8 20 4b 92 ae a5 02  ca
23d1 : f0 5e e0 01 f0 3c e0 02  c1
23d9 : f0 59 e0 03 f0 1a 78 a2  15
23e1 : 35 86 01 a0 00 51 fc 91  4f
23e9 : fc a4 02 88 20 4b 92 a0  b1
23f1 : 00 51 fc 91 fc 4c 71 c3  8b
23f9 : 78 a2 35 86 01 a0 00 11  18
2401 : fc 91 fc a4 02 88 20 4b  15
2409 : 92 a0 00 11 fc 91 fc 4c  f6
2411 : 71 c3 78 a2 35 86 01 a0  a3
2419 : 00 85 fe 11 fc 45 fe 91  d7
2421 : fc a4 02 88 20 4b 92 a0  e9
2429 : 00 11 fc 91 fc 4c 71 c3  a3
2431 : 4c 52 c3 78 a2 35 86 01  96
2439 : a0 00 11 fc 91 fc a4 02  55
2441 : 88 20 4b 92 a0 00 85 fe  1d
2449 : 11 fc 45 fe 91 fc 4c 71  1f
2451 : c3 78 a2 35 86 01 a0 00  93
2459 : 85 fe 11 fc 45 fe 91 fc  ce
2461 : a4 02 88 20 4b 92 a0 00  f8
2469 : 85 fe 11 fc 45 fe 91 fc  de
2471 : a2 37 86 01 58 a5 59 8d  a4
2479 : aa 02 a5 5a 8d ab 02 ad  73
2481 : aa 02 4a 4a a8 ad ac 02  b7
2489 : 4a 4a 4a aa b5 d9 29 03  b5
2491 : 09 80 85 63 29 03 09 e0  39
2499 : 85 5a 29 03 09 d8 85 fc  5d
24a1 : 29 03 09 e4 85 fe bd f0  54
24a9 : ec 85 62 85 59 85 fb 85  5e
24b1 : fd ad a6 02 0a 0a 0a 0a  9c
24b9 : 0d a7 02 91 62 91 59 ad  c0
24c1 : a8 02 91 fb 91 fd 60 a0  1a
24c9 : 01 8c b0 02 8c b1 02 8c  f4
24d1 : b7 02 88 8c b8 02 8c b2  70
24d9 : 02 8c b3 02 88 ad ba 02  33
24e1 : cd ab 02 90 0a d0 21 ad  1d
24e9 : b9 02 cd aa 02 b0 19 38  e6
24f1 : ad aa 02 ed b9 02 8d b4  7d
24f9 : 02 ad ab 02 ed ba 02 8d  d5
2501 : b5 02 8c b0 02 4c 1c c4  6d
2509 : 38 ad b9 02 ed aa 02 8d  1e
2511 : b4 02 ad ba 02 ed ab 02  cb
2519 : 8d b5 02 ad bb 02 cd ac  13
2521 : 02 b0 10 38 ad ac 02 ed  ab
2529 : bb 02 8d b6 02 8c b1 02  6f
2531 : 4c 3a c4 ed ac 02 8d b6  08
2539 : 02 ad b5 02 d0 24 ad b4  0e
2541 : 02 cd b6 02 b0 1c ae b6  2c
2549 : 02 8d b6 02 8e [illegible] [illegible] ad  f2
2551 : b0 02 8d b3 02 ad b1 02  34
2559 : 8d b2 02 c8 8c b0 02 8c  48
2561 : b1 02 ad b5 02 4a 8d ba  53
2569 : 02 ad b4 02 6a 8d b9 02  ad
2571 : 4c 17 c5 ad b0 02 30 12  70
2579 : 18 6d aa 02 8d aa 02 ad  c4
2581 : ab 02 69 00 8d ab 02 4c  5f
2589 : 9c c4 38 ad aa 02 e9 01  af
2591 : 8d aa 02 ad ab 02 e9 00  1c
2599 : 8d ab 02 18 ad ac 02 6d  a3
25a1 : b2 02 8d ac 02 18 ad b9  58
25a9 : 02 6d b6 02 8d b9 02 ad  5a
25b1 : ba 02 69 00 8d ba 02 ee  5b
25b9 : b7 02 d0 03 ee b8 02 ad  1e
25c1 : ba 02 cd b5 02 90 4f d0  2a
25c9 : 08 ad b4 02 cd b9 02 b0  29
25d1 : 45 38 ad b9 02 ed b4 02  3b
25d9 : 8d b9 02 ad ba 02 ed b5  58
25e1 : 02 8d ba 02 ad b3 02 30  7a
25e9 : 12 18 6d aa 02 8d aa 02  f3
25f1 : ad ab 02 69 00 8d ab 02  41
25f9 : 4c 0d c5 38 ad aa 02 e9  50
2601 : 01 8d aa 02 ad ab 02 e9  c8
2609 : 00 8d ab 02 18 ad ac 02  a1
2611 : 6d b1 02 8d ac 02 ae ac  78
2619 : 02 20 f4 c1 ad b8 02 cd  e5
2621 : b5 02 90 09 ad b7 02 cd  59
2629 : b4 02 90 01 60 4c 74 c4  e6
2631 : ad aa 02 8d b9 02 ad ab  1f
2639 : 02 8d ba 02 ad ac 02 8d  54
2641 : bb 02 4c 71 c5 ad bc 02  ff
2649 : 8d aa 02 ad bd 02 8d ab  db
2651 : 02 ad be 02 8d ac 02 60  21
2659 : 20 e3 c1 ad aa 02 8d b9  f5
2661 : 02 ad ab 02 8d ba 02 ad  77
2669 : ac 02 8d bb 02 20 fd ae  68
2671 : 20 e3 c1 ad aa 02 8d b7  09
2679 : 02 8d bc 02 ad ab 02 8d  0d
2681 : b8 02 8d bd 02 ad ac 02  9a
2689 : 8d be 02 20 c8 c3 4c 46  62
2691 : c5 20 15 fd 20 a3 fd 20  c3
2699 : 18 e5 a9 05 8d 18 d0 a9  df
26a1 : 01 8d 00 dd a9 80 8d 88  0a
26a9 : 02 a9 ce 8d 18 03 a9 89  38
26b1 : 8d 19 03 a9 1b 8d 11 d0  c5
26b9 : a9 00 8d aa 02 8d ab 02  5a
26c1 : 8d ac 02 8d a1 02 8d a2  7c
26c9 : 02 8d a3 02 8d a4 02 a9  14
26d1 : 01 8d a5 02 4c 44 e5 ad  1c
26d9 : b1 02 cd b4 02 90 26 d0  74
26e1 : 08 ad b0 02 cd b3 02 90  d0
26e9 : 1c ad b0 02 48 ad b3 02  0d
26f1 : 8d b0 02 68 8d b3 02 ad  3e
26f9 : b1 02 48 ad b4 02 8d b1  68
2701 : 02 68 8d b4 02 ad b2 02  8e
2709 : cd b5 02 90 0e ad b2 02  61
2711 : 48 ad b5 02 8d b2 02 68  25
2719 : 8d b5 02 60 20 e3 c1 ad  91
2721 : aa 02 8d b0 02 ad ab 02  86
2729 : 8d b1 02 ad ac 02 8d b2  3c
2731 : 02 20 fd ae 20 e3 c1 ad  1c
2739 : aa 02 8d b3 02 ad ab 02  fe
2741 : 8d b4 02 ad ac 02 8d b5  db
2749 : 02 20 d8 c5 ad b0 02 8d  ce
2751 : b6 02 ad b1 02 8d b7 02  19
2759 : ad b6 02 8d aa 02 ad b7  74
2761 : 02 8d ab 02 ad b2 02 8d  e9
2769 : ac 02 20 f4 c1 ad b5 02  21
2771 : 8d ac 02 20 f4 c1 ee b6  5f
2779 : 02 d0 03 ee b7 02 ad b7  34
2781 : 02 cd b4 02 90 d2 d0 08  ca
2789 : ad b6 02 cd b3 02 90 c8  eb
2791 : ad b0 02 8d aa 02 ad b1  9d
2799 : 02 8d ab 02 ad b2 02 8d  21
27a1 : ac 02 20 f4 c1 ad b3 02  51
27a9 : 8d aa 02 ad b4 02 8d ab  aa
27b1 : 02 20 f4 c1 ee b2 02 ad  20
27b9 : b2 02 cd b5 02 90 d1 60  43
27c1 : 20 e3 c1 ad aa 02 8d b0  4b
27c9 : 02 ad ab 02 8d b1 02 ad  97
27d1 : ac 02 8d b2 02 20 fd ae  ae
27d9 : 20 e3 c1 ad aa 02 8d b3  69
27e1 : 02 ad ab 02 8d b4 02 ad  c7
27e9 : ac 02 8d b5 02 20 d8 c5  c0
27f1 : ad b0 02 8d b6 02 ad b1  be
27f9 : 02 8d b7 02 ad b6 02 8d  a4
2801 : aa 02 ad b7 02 8d ab 02  4e
2809 : ad b2 02 8d ac 02 20 f4  87
2811 : c1 ee b6 02 d0 03 ee b7  87
2819 : 02 ad b7 02 cd b4 02 90  cc
2821 : db d0 7e ad b6 02 cd b3  d4
2829 : 02 90 d1 ad b0 02 8d b6  5c
2831 : 02 ad b1 02 8d b7 02 ee  33
2839 : b2 02 ad b2 02 cd b5 02  17
2841 : 90 ba 60 7b 0e fa 35 12  68
2849 : 40 a9 44 a0 c7 4c 28 ba  78
2851 : a9 e0 a0 9f 20 a2 bb 20  cd
2859 : 4a c7 20 64 e2 a9 ea a0  84
2861 : 9f 20 28 ba a9 c0 a0 02  99
2869 : 20 67 b8 20 f7 b7 a5 14  6b
2871 : 8d b9 02 a5 15 8d ba 02  bd
2879 : a9 e0 a0 9f 20 a2 bb 20  f5
2881 : 4a c7 20 6b e2 a9 ef a0  a1
2889 : 9f 20 28 ba a9 c6 a0 02  f1
2891 : 20 67 b8 20 f7 b7 a5 14  93
2899 : 8d bb 02 60 20 8a ad a2  e3
28a1 : c0 a0 02 20 d4 bb 20 f7  d2
28a9 : b7 20 fd ae 20 8a ad a2  18
28b1 : c6 a0 02 20 d4 bb 20 f7  e8
28b9 : b7 20 fd ae 20 8a ad a2  28
28c1 : ea a0 9f 20 d4 bb 20 f7  83
28c9 : b7 20 fd ae 20 8a ad a2  38
28d1 : ef a0 9f 20 d4 bb 20 f7  98
28d9 : b7 20 fd ae 20 8a ad a2  48
28e1 : e0 a0 9f 20 d4 bb 20 51  4b
28e9 : c7 a9 c0 a0 02 20 a2 bb  ec
28f1 : 20 f7 b7 a5 14 8d aa 02  0c
28f9 : a5 15 8d ab 02 a9 c6 a0  cb
2901 : 02 20 a2 bb 20 f7 b7 a5  1f
2909 : 14 8d ac 02 4c 74 c5 20  0f
2911 : 8a ad a2 c0 a0 02 20 d4  77
2919 : bb 20 f7 b7 20 fd ae 20  c6
2921 : 8a ad a2 c6 a0 02 20 d4  47
2929 : bb 20 f7 b7 20 fd ae 20  d6
2931 : 8a ad a2 ea a0 9f 20 d4  c9
2939 : bb 20 fd ae 20 8a ad a2  ac
2941 : ef a0 9f 20 d4 bb 20 fd  14
2949 : ae 20 8a ad a2 e0 a0 9f  52
2951 : 20 d4 bb 20 fd ae 20 8a  b9
2959 : ad a2 e5 a0 9f 20 d4 bb  aa
2961 : 20 fd ae 20 8a ad a2 f4  ba
2969 : a0 9f 20 d4 bb 20 f7 b7  87
2971 : 20 51 c7 ad b9 02 8d aa  19
2979 : 02 ad ba 02 8d ab 02 ad  da
2981 : bb 02 8d ac 02 20 f4 c1  af
2989 : a9 e0 a0 9f 20 a2 bb a9  18
2991 : f4 a0 9f 20 67 b8 a9 e5  70
2999 : a0 9f 20 5b bc 10 34 a2  df
29a1 : e0 a0 9f 20 d4 bb 20 51  0b
29a9 : c7 ad aa 02 ae b9 02 8d  0e
29b1 : b9 02 8e aa 02 ad ab 02  a4
29b9 : ae ba 02 8d ba 02 8e ab  44
29c1 : 02 ad ac 02 ae bb 02 8d  f1
29c9 : bb 02 8e ac 02 20 74 c5  3d
29d1 : 4c 89 c8 a9 e5 a0 9f 20  6b
29d9 : a2 bb a2 e0 a0 9f 20 d4  4f
29e1 : bb 20 51 c7 ad aa 02 ae  8f
29e9 : b9 02 8d b9 02 8e aa 02  81
29f1 : ad ab 02 ae ba 02 8d ba  32
29f9 : 02 8e ab 02 ad ac 02 ae  13
```

Listing des Sprite+Grafik-Basic. Bitte mit dem MSE eingeben.

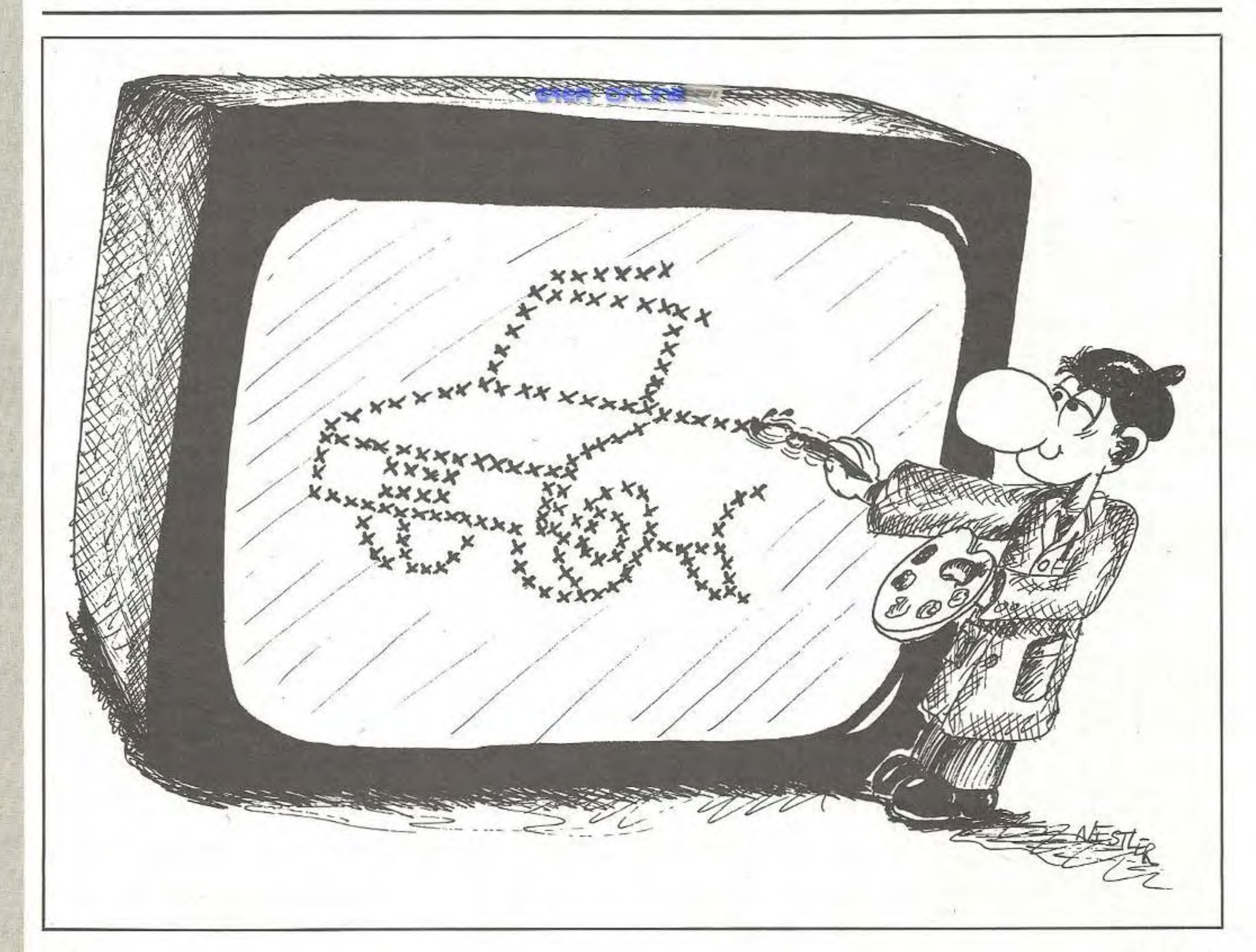
```
2a01 : bb 02 8d bb 02 8e ac 02   e3
2a09 : 4c 74 c5 85 57 a9 00 85   7f
2a11 : 58 a0 03 06 58 18 06 57   48
2a19 : 90 02 e6 58 88 d0 f4 a5   9d
2a21 : 58 09 d0 85 58 ad a3 02   68
2a29 : 29 80 f0 06 a5 58 09 08   e1
2a31 : 85 58 a9 00 8d b0 02 8d   ce
2a39 : b1 02 ac b0 02 20 4b 92   a0
2a41 : a2 33 86 01 78 ac b1 02   f6
2a49 : 31 57 d0 1f ee b0 02 ad   16
2a51 : b0 02 c9 08 90 e4 a9 00   4d
2a59 : 8d b0 02 ee b1 02 ad b1   e2
2a61 : 02 c9 08 90 d5 a2 37 86   b8
2a69 : 01 58 60 ad b0 02 49 07   b2
2a71 : 8d bc 02 ad a3 02 29 0f   a0
2a79 : a8 a9 00 18 6d bc 02 88   cf
2a81 : 10 f9 18 6d aa 02 8d b9   a6
2a89 : 02 ad ab 02 69 00 8d ba   cf
2a91 : 02 ad a4 02 29 0f a8 a9   d4
2a99 : 00 18 6d b1 02 88 10 f9   cf
2aa1 : 18 6d ac 02 8d bb 02 ad   f5
2aa9 : aa 02 8d b6 02 ad ab 02   cf
2ab1 : 8d b7 02 ad ac 02 8d b8   d3
2ab9 : 02 a9 00 8d b4 02 8d b5   3f
2ac1 : 02 ad b9 02 18 6d b4 02   0c
2ac9 : 8d aa 02 ad ba 02 69 00   43
2ad1 : 8d ab 02 ad bb 02 18 6d   71
2ad9 : b5 02 8d ac 02 20 f4 c1   01
2ae1 : ee b4 02 ad a3 02 29 0f   6d
2ae9 : cd b4 02 b0 d4 a9 00 8d   5d
2af1 : b4 02 ee b5 02 ad a4 02   3d
2af9 : 29 0f cd b5 02 b0 c2 ad   e0
2b01 : b6 02 8d aa 02 ad b7 02   e1
2b09 : 8d ab 02 ad b8 02 8d ac   cd
2b11 : 02 a2 33 78 86 01 4c 4d   7c
2b19 : c9 ad a4 02 29 0f 18 69   60
2b21 : 01 0a 0a 0a 18 6d ac 02   8f
2b29 : 8d ac 02 60 ad a4 02 29   f3
2b31 : 0f 18 69 01 0a 0a 0a 85   eb
2b39 : 02 ad ac 02 38 e5 02 8d   53
2b41 : ac 02 60 ad a3 02 29 0f   c9
2b49 : 18 69 01 0a 0a 0a 85 02   a3
2b51 : ad aa 02 38 e5 02 8d aa   d5
2b59 : 02 ad ab 02 e9 00 8d ab   89
2b61 : 02 30 01 60 ad a2 02 d0   61
2b69 : 10 a9 40 8d aa 02 a9 01   73
2b71 : 8d ab 02 20 71 ca 4c 88   08
2b79 : ca a9 a0 8d aa 02 a9 00   53
2b81 : 8d ab 02 20 71 ca 4c 88   18
2b89 : ca ad a3 02 29 0f 18 69   91
2b91 : 01 0a 0a 0a 85 02 18 6d   ff
2b99 : aa 02 8d aa 02 ad ab 02   3d
2ba1 : 69 00 8d ab 02 a5 02 18   69
2ba9 : 6d aa 02 8d b0 02 ad ab   c7
2bb1 : 02 69 00 8d b1 02 ad a2   41
2bb9 : 02 d0 18 ad b1 02 f0 12   f2
2bc1 : ad b0 02 c9 41 90 0b a9   98
2bc9 : 00 8d aa 02 8d ab 02 4c   52
2bd1 : 1a ca 60 ad b1 02 d0 ef   6c
2bd9 : ad b0 02 c9 a1 b0 e8 60   9c
2be1 : c9 40 90 17 c9 60 b0 06   40
2be9 : 38 e9 40 4c fc ca c9 c0   7e
2bf1 : b0 06 38 e9 40 4c fc ca   e0
2bf9 : 38 e9 80 48 ad a3 02 29   a1
2c01 : 40 f0 06 68 09 80 4c 0b   24
2c09 : cb 68 4c 0c c9 20 9e ad   10
2c11 : 20 a3 b6 c9 00 f0 24 85   0d
2c19 : 14 a0 00 84 15 a4 15 b1   3c
2c21 : 22 c9 20 90 17 c9 a0 b0   e6
2c29 : 04 c9 80 b0 0f 20 e1 ca   57
2c31 : 20 8a ca e6 15 a5 15 c5   84
2c39 : 14 90 e2 60 c9 11 d0 06   ce
2c41 : 20 1a ca 4c 34 cb c9 1d   ad
2c49 : d0 06 20 8a ca 4c 34 cb   ed
2c51 : c9 91 d0 06 20 2d ca 4c   07
2c59 : 34 cb c9 9d d0 06 20 44   df
2c61 : ca 4c 34 cb c9 0d d0 06   2c
2c69 : 20 c8 ca 4c 34 cb c9 0e   0e
2c71 : d0 0b ad a3 02 09 80 8d   2c
2c79 : a3 02 4c 34 cb c9 12 d0   ac
2c81 : 0b ad a3 02 09 40 8d a3   9c
2c89 : 02 4c 34 cb c9 8d f0 d8   b6
2c91 : c9 8e d0 0b ad a3 02 29   89
2c99 : 7f 8d a3 02 4c 34 cb c9   31
2ca1 : 92 d0 0b ad a3 02 29 bf   82
2ca9 : 8d a3 02 4c 34 cb c9 93   02
2cb1 : d0 06 20 70 c0 4c bd cb   97
2cb9 : c9 13 d0 0e a9 00 8d aa   28
2cc1 : 02 8d ab 02 8d ac 02 4c   94
2cc9 : 34 cb c9 01 d0 0e ee a3   f6
2cd1 : 02 ad a3 02 29 cf 8d a3   62
2cd9 : 02 4c 34 cb c9 02 d0 06   84
2ce1 : ce a3 02 4c d2 cb c9 03   44
2ce9 : d0 0e ee a4 02 ad a4 02   35
2cf1 : 29 0f 8d a4 02 4c 34 cb   85
2cf9 : c9 04 d0 06 ce a4 02 4c   6c
2d01 : ee cb c9 06 d0 08 a9 00   fc
2d09 : 8d a5 02 4c 34 cb c9 07   4a
2d11 : d0 05 a9 01 4c 09 cc c9   c2
2d19 : 0a d0 05 a9 02 4c 09 cc   42
2d21 : c9 0b d0 05 a9 03 4c 09   3b
2d29 : cc c9 0c d0 05 a9 04 4c   3d
2d31 : 09 cc c9 15 d0 06 ee ac   08
2d39 : 02 4c 34 cb c9 0f d0 06   4c
2d41 : ce ac 02 4c 34 cb c9 10   58
2d49 : d0 0b ee aa 02 d0 03 ee   40
2d51 : ab 02 4c 34 cb c9 16 d0   9c
2d59 : 10 ce aa 02 ad aa 02 c9   87
2d61 : ff d0 03 ce ab 02 4c 34   c7
2d69 : cb c9 05 d0 05 a9 01 4c   ae
2d71 : c6 cc c9 1c d0 05 a9 02   73
2d79 : 4c c6 cc c9 1e d0 05 a9   64
2d81 : 05 4c c6 cc c9 1f d0 05   da
2d89 : a9 06 4c c6 cc c9 90 d0   20
2d91 : 05 a9 00 4c c6 cc c9 9c   28
2d99 : d0 05 a9 04 4c c6 cc c9   98
2da1 : 9e d0 05 a9 07 4c c6 cc   a5
2da9 : c9 9f d0 05 a9 03 4c c6   88
2db1 : cc c9 81 d0 05 a9 08 4c   33
2db9 : c6 cc c9 95 90 1a c9 9c   45
2dc1 : b0 16 38 e9 8c 48 ad a5   d5
2dc9 : 02 f0 0c c9 04 f0 08 aa   bd
2dd1 : 68 9d a5 02 4c 34 cb 68   18
2dd9 : 4c 34 cb 20 eb b7 ad a2   af
2de1 : 02 d0 30 a5 14 8d aa 02   68
2de9 : a5 15 8d ab 02 8e ac 02   3d
2df1 : 20 86 c1 a8 20 4b 92 78   71
2df9 : a2 35 86 01 a0 00 31 fc   c0
2e01 : a2 37 86 01 58 c9 00 f0   b6
2e09 : 05 a0 01 4c a2 b3 a0 00   72
2e11 : 4c a2 b3 8e ac 02 18 06   b4
2e19 : 14 90 02 e6 15 a5 14 8d   bd
2e21 : aa 02 a5 15 8d ab 02 20   57
2e29 : 86 c1 85 02 a8 20 4b 92   0f
2e31 : 78 a2 35 86 01 a0 00 31   90
2e39 : fc f0 05 a2 02 4c 43 cd   6e
2e41 : a2 00 a4 02 88 20 4b 92   29
2e49 : 31 fc f0 01 e8 8a a8 a2   1f
2e51 : 37 86 01 58 4c a2 b3 20   00
2e59 : 9e b7 20 ba ff 20 fd ae   89
2e61 : 20 9e ad 20 a3 b6 a6 22   0e
2e69 : a4 23 20 bd ff a2 00 86   81
2e71 : fb a9 a0 85 fc a0 36 84   d0
2e79 : 01 a0 c0 a9 fb 20 d8 ff   54
2e81 : a9 37 85 01 a9 00 85 fb   f0
2e89 : 85 fd a9 87 85 fc a9 e3   17
2e91 : 85 fe 78 a9 35 85 01 a0   ad
2e99 : f7 b1 fd 91 fb 88 c0 ff   21
2ea1 : d0 f7 c6 fe c6 fc a5 fc   e3
2ea9 : c9 84 b0 ed a9 37 85 01   0a
2eb1 : 58 ea ea ea ea ea ea ea   1e
2eb9 : ea ea a9 84 85 fc a2 00   de
2ec1 : 86 fb a0 88 a9 fb 20 d8   2b
2ec9 : ff a9 00 85 fb 85 fd a9   85
2ed1 : 87 85 fc a9 e7 85 fe 78   27
2ed9 : a9 35 85 01 a0 f7 b1 fd   2b
2ee1 : 91 fb 88 c0 ff d0 f7 c6   9e
2ee9 : fe c6 fc a5 fc c9 84 b0   d0
2ef1 : ed a9 37 85 01 58 ea ea   86
2ef9 : ea ea ea ea ea ea ea a9   75
2f01 : 84 85 fc a2 00 86 fb a0   41
2f09 : 88 a9 fb 20 d8 ff 60 20   b8
2f11 : 9e b7 a0 00 20 ba ff 20   cb
2f19 : fd ae 20 9e ad 20 a3 b6   21
2f21 : a6 22 a4 23 20 bd ff a9   a9
2f29 : 00 a2 00 a0 a0 20 d5 ff   f1
2f31 : a9 00 a2 00 a0 00 20 bd   89
2f39 : ff a9 00 a2 00 a0 e0 20   2a
2f41 : d5 ff a9 00 a2 00 a0 00   2d
2f49 : 20 bd ff a9 00 a2 00 a0   d3
2f51 : e4 4c d5 ff ff 00 00 ff   d1
2f59 : ff 00 00 ff ff 00 00 ff   58
2f61 : ff 00 00 ff ff 00 00 ff   60
2f69 : ff 00 00 ff ff 00 00 ff   68
2f71 : ff 00 00 ff ff 00 00 ff   70
2f79 : ff 00 00 ff ff 00 80 ff   7a
2f81 : ff 00 00 ff ff 00 00 ff   80
2f89 : ff 00 00 ff ff 00 00 ff   88
2f91 : ff 00 00 ff ff 00 00 ff   90
2f99 : ff 00 00 ff ff 00 00 ff   98
2fa1 : ff 00 00 ff ff 00 00 ff   a0
2fa9 : ff 00 00 ff ff 00 00 ff   a8
2fb1 : ff 00 00 ff ff 00 00 ff   b0
2fb9 : ff 00 00 ff ff 00 00 ff   b8
2fc1 : ff 00 00 ff ff 00 00 ff   c0
2fc9 : ff 00 00 ff ff 00 00 ff   c8
2fd1 : ff 00 00 ff ff 00 00 ff   d0
2fd9 : ff 00 00 ff ff 00 00 ff   d8
2fe1 : ff 00 00 ff ff 00 00 ff   e0
2fe9 : ff 00 00 ff ff 00 00 ff   e8
2ff1 : ff 00 00 ff ff 00 00 ff   f0
2ff9 : ff 00 00 ff ff 00 02 00   00
3001 : 00 ff ff 00 00 ff ff 00   01
3009 : 00 ff ff 00 00 ff ff 00   09
3011 : 00 ff ff 00 00 ff ff 00   11
3019 : 00 ff ff 00 00 ff ff 00   19
3021 : 00 ff ff 00 00 ff ff 00   21
3029 : 00 ff ff 00 00 ff ff 00   29
3031 : 00 ff ff 00 00 ff ff 00   31
3039 : 00 ff ff 00 00 ff ff 00   39
3041 : 00 ff ff 00 00 ff df 00   c0
3049 : 00 ff ff 00 00 ff ff 00   49
3051 : 00 ff ff 00 04 ff ff 00   91
3059 : 00 ff ff 00 00 ff ff 00   59
3061 : 00 ff ff 00 00 ff ff 00   61
3069 : 00 ff ff 00 00 ff ff 00   69
3071 : 00 ff ff 00 00 ff ff 00   71
3079 : 00 ff ff 00 00 ff ff 00   79
3081 : 00 ff ff 00 00 ff ff 00   81
3089 : 00 ff ff 00 00 ff ff 00   89
3091 : 00 ff ff 00 00 ff ff 00   91
3099 : 00 ff ff 00 00 ff ff 00   99
30a1 : 00 ff ff 00 00 ff ff 00   a1
30a9 : 00 ff ff 00 00 ff ff 00   a9
30b1 : 00 ff ff 00 00 ff ff 00   b1
30b9 : 00 ff ff 00 00 ff ff 00   b9
30c1 : 00 ff ff 00 00 ff ff 00   c1
30c9 : 00 ff ff 00 00 ff ff 00   c9
30d1 : 00 ff ff 00 00 ff ff 00   d1
30d9 : 00 ff ff 00 00 ff ff 00   d9
30e1 : 00 ff ff 00 00 ff ff 00   e1
30e9 : 00 ff ff 00 00 ff ff 00   e9
30f1 : 00 ff ff 00 00 ff ff 00   f1
30f9 : 00 ff ff 00 00 ff 7d ff   ef
3101 : ff 00 00 ff ff 00 00 ff   00
3109 : ff 00 00 ff ff 00 00 ff   08
3111 : ff 00 00 ff ff 00 00 ff   10
3119 : ff 00 00 ff ff 00 00 ff   18
3121 : ff 00 00 ff ff 00 00 ff   20
3129 : ff 00 00 ff ff 00 00 ff   28
3131 : ff 00 00 ff ff 00 00 ff   30
3139 : ff 00 00 ff ff 00 00 ff   38
3141 : ff 00 00 ff ff 00 00 ff   40
3149 : ff 00 00 ff ff 00 00 ff   48
3151 : ff 00 00 ff ff 00 00 ff   50
3159 : ff 00 00 ff ff 00 00 ff   58
3161 : ff 00 00 ff ff 00 00 ff   60
3169 : ff 00 00 ff ff 00 00 ff   68
3171 : ff 00 00 ff ff 00 00 ff   70
3179 : ff 00 00 ff ff 00 00 ff   78
3181 : ff 00 00 ff ff 00 00 ff   80
3189 : ff 00 00 ff ff 00 00 ff   88
3191 : ff 00 00 ff ff 00 00 ff   90
3199 : ff 00 00 ff ff 00 00 ff   98
31a1 : ff 00 00 ff ff 00 00 ff   a0
31a9 : ff 00 00 ff ff 00 00 ff   a8
31b1 : ff 00 00 ff ff 00 00 ff   b0
31b9 : ff 00 00 ff ff 00 00 ff   b8
31c1 : ff 00 00 ff ff 00 00 ff   c0
31c9 : ff 00 00 ff ff 00 00 ff   c8
31d1 : ff 00 00 ff ff 80 00 ff   d4
31d9 : ff 00 00 ff ff 00 00 ff   d8
31e1 : ff 00 00 ff ff 00 00 ff   e0
31e9 : ff 00 00 ff ff 00 00 ff   e8
31f1 : ff 00 00 ff ff 00 00 ff   f0
31f9 : ff 00 00 ff ff 00 20 78   6a
3201 : a9 0d 8d ec 8e a9 71 8d   49
3209 : ed 8e 58 60 a5 cb c9 03   45
3211 : b0 03 4c 31 ea c9 07 b0   f7
3219 : f9 48 a5 9d d0 07 a5 11   51
3221 : f0 03 68 d0 ed 68 38 e9   9e
3229 : 03 8d ff 70 0a 0a 0a 0a   2e
3231 : 8d fe 70 ad ff 70 0a 0a   cf
3239 : 18 6d fe 70 8d ff 70 ad   cb
3241 : 8d 02 c9 03 b0 cc 8d fe   48
3249 : 70 0a 0a 0a 0a 18 6d ff   99
3251 : 70 8d ff 70 ad fe 70 0a   3e
3259 : 0a 0a 0a 0a 0a 18 6d ff   43
3261 : 70 8d 6e 71 a9 70 8d 6f   95
3269 : 71 a2 ff e8 bd 00 70 f0   c8
3271 : 06 20 d2 ff 4c 6c 71 4c   c2
3279 : 31 ea 20 9e b7 8e fb 70   bc
3281 : 20 fd ae 20 9e ad 20 a3   6f
3289 : b6 c9 15 90 03 4c 58 b6   dd
3291 : 8d fa 70 ad fb 70 c9 01   d9
3299 : f0 51 c9 09 f0 4d c9 0a   7a
32a1 : f0 49 c9 0b f0 45 c9 02   6e
32a9 : d0 05 a9 05 4c ec 71 c9   8c
32b1 : 03 d0 05 a9 02 4c ec 71   ac
32b9 : c9 04 d0 05 a9 06 4c ec   2f
32c1 : 71 c9 05 d0 05 a9 03 4c   b4
32c9 : ec 71 c9 06 d0 05 a9 07   8b
32d1 : 4c ec 71 c9 07 d0 05 a9   87
32d9 : 00 4c ec 71 c9 08 d0 05   93
32e1 : a9 04 4c ec 71 c9 0c d0   74
32e9 : 02 a9 08 8d ff 70 0a 0a   33
32f1 : 8d fe 70 ad ff 70 0a 0a   8f
32f9 : 0a 0a 18 6d fe 70 85 14   6e
3301 : a9 70 85 15 ac fa 70 a9   9e
3309 : 00 91 14 88 30 07 b1 22   2e
3311 : 91 14 88 10 f9 60 a2 08   0e
3319 : a0 00 b1 5f 85 45 c8 b1   1b
3321 : 5f 85 60 a5 45 85 5f ca   a3
3329 : 10 ee a0 02 b1 5f 85 14   6d
3331 : 8d d0 02 c8 b1 5f 85 15   16
3339 : 8d d1 02 60 ad d0 02 85   b0
3341 : 14 ad d1 02 85 15 20 13   88
3349 : a6 a5 5f 8d d2 02 a5 60   e0
3351 : 8d d3 02 20 17 72 ad d2   ae
3359 : 02 85 5f ad d3 02 85 60   cf
3361 : 4c c9 a6 20 44 e5 20 8a   48
3369 : ad 20 f7 b7 a5 14 8d d0   ee
3371 : 02 a5 15 8d d1 02 4c 3d   16
3379 : 72 ee d0 02 d0 03 ee d1   5b
3381 : 02 20 44 e5 4c 3d 72 ad   35
3389 : d0 02 85 14 ad d1 02 85   bb
```

```
3391 : 15 20 13 a6 a5 5f 8d d2  81
3399 : 02 a5 60 8d d3 02 38 a5  b1
33a1 : 5f e9 60 85 5f a5 60 e9  36
33a9 : 00 85 60 a0 ff c8 a5 60  36
33b1 : c9 04 b0 09 a9 00 8d d0  3c
33b9 : 02 8d d1 02 60 b1 5f d0  e9
33c1 : ec c8 b1 5f cd d2 02 d0  87
33c9 : e4 c8 b1 5f cd d3 02 d0  8f
33d1 : dc c8 b1 5f 8d d0 02 c8  63
33d9 : b1 5f 8d d1 02 60 a2 12  a9
33e1 : 8a 48 20 88 72 ad d0 02  84
33e9 : 0d d1 02 f0 05 68 aa ca  51
33f1 : 10 ee 4c 82 72 a5 2b a4  26
33f9 : 2c a0 03 c8 b1 22 d0 fb  b7
3401 : c8 98 18 69 22 a0 00 91  93
3409 : 2b a5 23 69 00 c8 91 2b  e0
3411 : 20 33 a5 a5 22 69 02 85  69
3419 : 2d a5 23 69 00 85 2e 20  34
3421 : 63 a6 4c 7b e3 a9 01 85  f4
3429 : 02 20 44 e5 a9 46 20 d2  fc
3431 : ff a9 4b 20 d2 ff a9 45  3a
3439 : 20 d2 ff a9 59 20 d2 ff  d9
3441 : a6 02 a9 00 20 cd bd a9  0d
3449 : 2c 20 d2 ff a9 22 20 d2  0c
3451 : ff a5 02 c9 01 d0 05 a9  db
3459 : 01 4c a2 73 c9 02 d0 05  91
3461 : a9 05 4c a2 73 c9 03 d0  27
3469 : 05 a9 02 4c a2 73 c9 04  42
3471 : d0 05 a9 06 4c a2 73 c9  2a
3479 : 05 d0 05 a9 03 4c a2 73  61
3481 : c9 06 d0 05 a9 07 4c a2  6b
3489 : 73 c9 07 d0 05 a9 00 4c  f3
3491 : a2 73 c9 08 d0 05 a9 04  44
3499 : 4c a2 73 c9 0c d0 02 a9  ef
34a1 : 08 48 0a 0a 85 14 68 0a  40
34a9 : 0a 0a 0a 18 65 14 aa bd  5b
34b1 : 00 70 f0 07 20 d2 ff e8  71
34b9 : 4c b0 73 a9 22 20 d2 ff  de
34c1 : a9 0d 20 d2 ff a5 02 c9  1c
34c9 : 0c b0 2d e6 02 4c 2d 73  74
34d1 : a5 2d a4 2e 85 14 84 15  31
34d9 : c4 30 d0 02 c5 2f b0 18  f2
34e1 : 69 02 90 01 c8 85 22 84  da
34e9 : 23 20 17 74 20 4b 74 8a  b4
34f1 : 10 07 20 54 74 4c 06 74  c2
34f9 : 60 98 30 06 20 64 74 4c  02
3501 : 06 74 20 6d 74 a9 0d 20  00
3509 : d2 ff a5 14 a4 15 18 69  ed
3511 : 07 90 c1 c8 b0 be a0 00  6d
3519 : b1 14 aa 29 7f 20 d2 ff  e8
3521 : c8 b1 14 a8 29 7f f0 03  34
3529 : 20 d2 ff 8a 10 11 98 30  50
3531 : 0a a9 21 20 d2 ff 68 68  fc
3539 : 4c 06 74 a9 25 d0 4e 98  1e
3541 : 10 04 a9 24 d0 47 60 20  4b
3549 : d2 ff a9 20 20 d2 ff a9  75
3551 : 3d d0 3a a0 00 b1 22 aa  04
3559 : c8 b1 22 a8 8a 20 95 b3  ff
3561 : 4c 67 74 20 a6 bb 20 dd  06
3569 : bd 4c 1e ab 20 8c 74 a0  c3
3571 : 02 b1 22 85 25 88 b1 22  27
3579 : 85 24 88 b1 22 85 26 f0  31
3581 : 0a b1 24 20 d2 ff c8 c4  4b
3589 : 26 d0 f6 a9 22 4c d2 ff  da
3591 : a5 2d 38 e9 02 85 2b b0  72
3599 : 02 c6 2e a5 2e 85 2c a9  52
35a1 : 00 85 0a 20 d4 e1 a5 0a  f1
35a9 : a6 2b a4 2c 20 d5 ff 90  65
35b1 : 03 4c d1 e1 a9 01 85 2b  9a
35b9 : a9 04 85 2c 4c 95 e1 20  84
35c1 : 8a ad 20 f7 b7 20 13 a6  3f
35c9 : 90 0e a5 5f 85 2b a5 60  bf
35d1 : 85 2c 20 fd ae 4c a0 74  ed
35d9 : a2 11 4c 37 a4 20 9e b7  33
35e1 : 20 ba ff 20 fd ae 20 9e  75
35e9 : ad 20 a3 b6 a6 22 a4 23  ba
35f1 : 20 bd ff 20 fd ae 20 8a  df
35f9 : ad 20 f7 b7 20 13 a6 90  01
3601 : d7 a5 5f 85 fb a5 60 85  ad
3609 : fc 20 fd ae 20 8a ad 20  b8
3611 : f7 b7 20 13 a6 90 c1 a0  86
3619 : 00 b1 5f 85 45 c8 b1 5f  9a
3621 : 85 60 a5 45 85 5f a0 00  be
3629 : b1 5f 85 14 a9 00 91 5f  0d
3631 : c8 b1 5f 85 15 a9 00 91  1c
3639 : 5f a5 5f 18 69 02 aa 90  b8
3641 : 02 e6 60 a4 60 a9 fb 20  e6
3649 : d8 ff a5 14 a0 00 91 5f  1c
3651 : c8 a5 15 91 5f 60 a9 00  03
3659 : 85 02 ea ea ea a2 00 bd  36
3661 : 3c 8a 29 7f 20 d2 ff bd  31
3669 : 3c 8a 29 80 d0 04 e8 4c  ae
3671 : 60 75 a9 20 20 d2 ff a9  e6
3679 : 00 20 d2 ff e6 02 a5 02  57
3681 : c9 06 b0 09 e8 bd 3c 8a  1d
3689 : f0 10 4c 60 75 a9 00 85  50
3691 : 02 a9 0d 20 d2 ff e8 4c  19
3699 : 86 75 a9 00 85 02 a2 00  37
36a1 : bd 3c 8b 29 7f 20 d2 ff  c9
36a9 : bd 3c 8b 29 80 d0 04 e8  fd
36b1 : 4c a1 75 a9 20 20 d2 ff  af
36b9 : a9 00 20 d2 ff e6 02 a5  4f
36c1 : 02 c9 06 b0 09 e8 bd 3c  87
36c9 : 8b f0 10 4c a1 75 a9 00  c6
36d1 : 85 02 a9 0d 20 d2 ff e8  ce
36d9 : 4c c7 75 60 a9 00 85 02  27
36e1 : a2 00 bd 9e a0 29 7f 20  58
36e9 : d2 ff bd 9e a0 29 80 d0  f5
36f1 : 04 e8 4c e3 75 a9 20 20  5e
36f9 : d2 ff a9 00 20 d2 ff e6  9c
3701 : 02 a5 02 c9 07 b0 09 e8  7c
3709 : bd 9e a0 f0 10 4c e3 75  39
3711 : a9 00 85 02 a9 0d 20 d2  85
3719 : ff e8 4c 09 76 60 a0 01  b0
3721 : b1 5f 85 23 a5 2d 85 22  66
3729 : a5 60 85 25 a5 5f 88 f1  60
3731 : 5f 18 65 2d 85 2d 85 24  bb
3739 : a5 2e 69 ff 85 2e e5 60  72
3741 : aa 38 a5 5f e5 2d a8 b0  28
3749 : 03 e8 c6 25 18 65 22 90  6d
3751 : 03 c6 23 18 b1 22 91 24  3e
3759 : c8 d0 f9 e6 23 e6 25 ca  78
3761 : d0 f2 4c 33 a5 20 8a ad  05
3769 : 20 f7 b7 a5 14 8d b0 02  9c
3771 : a5 15 8d b1 02 20 fd ae  b1
3779 : 20 8a ad 20 f7 b7 a5 14  4a
3781 : 8d b2 02 a5 15 8d b3 02  2d
3789 : ad b0 02 85 14 ad b1 02  39
3791 : 85 15 20 13 a6 90 03 20  47
3799 : 1f 76 ee b0 02 d0 03 ee  56
37a1 : b1 02 ad b1 02 cd b3 02  56
37a9 : 90 de d0 08 ad b0 02 cd  e1
37b1 : b2 02 90 d4 60 42 41 53  e7
37b9 : 49 43 20 3a 20 00 56 41  d1
37c1 : 52 49 41 42 4c 45 20 3a  34
37c9 : 20 00 41 52 52 41 59 53  bf
37d1 : 20 3a 20 00 a2 ff e8 bd  60
37d9 : b6 76 20 d2 ff d0 f7 a5  de
37e1 : 2e 38 e5 2c aa a5 2d 38  27
37e9 : e5 2b b0 01 ca 48 8a a8  1a
37f1 : 68 aa 98 20 cd bd a9 0d  64
37f9 : 20 d2 ff a2 ff e8 bd bf  94
3801 : 76 20 d2 ff d0 f7 a5 30  00
3809 : 38 e5 2e aa a5 2f 38 e5  95
3811 : 2d b0 01 ca 48 8a a8 68  7c
3819 : aa 98 20 cd bd a9 0d 20  6f
3821 : d2 ff a2 ff e8 bd cb 76  34
3829 : 20 d2 ff d0 f7 a5 32 38  b2
3831 : e5 30 aa a5 31 38 e5 2f  58
3839 : b0 01 ca 48 8a a8 68 aa  0a
3841 : 98 4c cd bd a2 ff e8 bd  74
3849 : 00 02 c9 cd d0 f8 e8 bd  6a
3851 : 00 02 c9 23 d0 f1 [illegible] a0  ab
3859 : 00 bd 00 02 99 00 02 f0  fc
3861 : 05 e8 c8 4c 5a 77 a9 03  a4
3869 : 85 fb a9 04 85 fc a0 00  99
3871 : 20 84 77 b1 fb f0 21 20  f3
3879 : b9 77 e6 fb d0 02 e6 fc  da
3881 : 4c 74 77 b1 fb 85 14 e6  25
3889 : fb d0 02 e6 fc b1 fb 85  a2
3891 : 15 e6 fb d0 02 e6 fc 60  3e
3899 : c8 b1 fb f0 13 a0 00 a5  d8
38a1 : fb 18 69 03 85 fb 90 02  e2
38a9 : e6 fc 20 84 77 4c 74 77  41
38b1 : c8 b1 fb d0 e8 6c 02 a0  a6
38b9 : a0 00 b9 00 02 f0 0b d1  3f
38c1 : fb d0 04 c8 4c bb 77 a0  00
38c9 : 00 60 a5 14 c5 fd d0 07  83
38d1 : a5 15 c5 fe d0 01 60 a6  36
38d9 : 14 86 fd a5 15 85 fe 20  1e
38e1 : cd bd a9 20 20 d2 ff e6  62
38e9 : 02 a5 02 c9 06 90 09 a9  d4
38f1 : 0d 20 d2 ff a9 00 85 02  78
38f9 : a0 00 60 a9 00 85 02 a9  6e
3901 : ff 85 fd 85 fe 4c 45 77  49
3909 : 20 34 78 86 7a 84 7b a0  2d
3911 : ff c8 b9 00 02 f0 0f c9  5a
3919 : 3f 90 f6 20 73 00 aa a2  89
3921 : ff 86 3a 4c 9c a4 a9 83  18
3929 : 8d 02 03 a9 a4 8d 03 03  76
3931 : 4c 80 a4 ae c0 02 86 14  1b
3939 : ad c1 02 85 15 20 13 a6  e4
3941 : 90 08 a9 02 8d 20 d0 4c  36
3949 : 50 78 a9 0e 8d 20 d0 ae  7c
3951 : c0 02 ad c1 02 85 62 86  99
3959 : 63 a2 90 38 20 49 bc 20  b8
3961 : df bd a2 00 bd 00 01 f0  89
3969 : 0a 9d 00 02 20 d2 ff e8  ed
3971 : 4c 65 78 ea a9 20 20 d2  ad
3979 : ff 20 12 e1 c9 0d f0 17  40
3981 : 9d 00 02 e8 e0 59 90 f1  bb
3989 : a9 83 8d 02 03 a9 a4 8d  c3
3991 : 03 03 a2 17 20 37 a4 ad  4b
3999 : c0 02 18 6d c2 02 8d c0  02
39a1 : 02 ad c1 02 6d c3 02 8d  43
39a9 : c1 02 4c ca aa 20 8a ad  09
39b1 : 20 f7 b7 a5 14 8d c0 02  24
39b9 : a5 15 8d c1 02 20 fd ae  fb
39c1 : 20 8a ad 20 f7 b7 a5 14  92
39c9 : 8d c2 02 a5 15 8d c3 02  bd
39d1 : a9 09 8d 02 03 a9 78 8d  1d
39d9 : 03 03 4c 80 a4 a2 0f a9  70
39e1 : 00 9d b0 02 ca 10 fa 20  76
39e9 : 79 00 8d b0 02 d0 05 a2  dc
39f1 : 12 4c 37 a4 20 73 00 8d  44
39f9 : b1 02 f0 46 c9 24 d0 08  c1
3a01 : a9 80 8d b1 02 4c 45 79  0e
3a09 : c9 25 d0 10 a9 80 8d b1  d3
3a11 : 02 ad b0 02 09 80 8d b0  82
3a19 : 02 4c 45 79 20 73 00 f0  41
3a21 : 23 c9 24 d0 0b ad b1 02  35
3a29 : 09 80 8d b1 02 4c 45 79  96
3a31 : c9 25 d0 bb ad b0 02 09  b3
3a39 : 80 8d b0 02 ad b1 02 09  6f
3a41 : 80 8d b1 02 a5 2f 85 fb  16
3a49 : a5 30 85 fc a5 fc c5 32  c5
3a51 : 90 09 d0 06 a5 fb c5 31  0e
3a59 : 90 01 60 a0 00 b1 fb cd  af
3a61 : b0 02 d0 08 c8 b1 fb cd  ed
3a69 : b1 02 f0 1b a0 02 b1 fb  94
3a71 : 85 fd c8 b1 fb 85 fe a5  90
3a79 : fb 18 65 fd 85 fb a5 fe  66
3a81 : 65 fc 85 fc 4c 4d 79 a9  ce
3a89 : 00 8d b2 02 ad b1 02 29  ff
3a91 : 80 f0 03 ee b2 02 ad b0  7b
3a99 : 02 29 80 f0 03 ee b2 02  e4
3aa1 : a0 04 b1 fb 8d b3 02 c8  3f
3aa9 : 0a aa b1 fb 99 af 02 c8  a5
3ab1 : ca 10 f7 ad b3 02 0a aa  00
3ab9 : a9 00 9d b4 02 a9 01 9d  0d
3ac1 : b5 02 e8 e8 e0 06 90 f0  31
3ac9 : a2 05 a9 00 9d ba 02 ca  a5
3ad1 : 10 fa ad b3 02 0a 18 69  e4
3ad9 : 05 18 65 fb 85 fb a5 fc  8c
3ae1 : 69 00 85 fc 20 41 7a ee  1f
3ae9 : bf 02 d0 03 ee be 02 ad  86
3af1 : be 02 cd b8 02 d0 ed ad  f4
3af9 : bf 02 cd b9 02 d0 e5 a9  f5
3b01 : 00 8d be 02 8d bf 02 ee  74
3b09 : bd 02 d0 03 ee bc 02 ad  94
3b11 : bc 02 cd b6 02 d0 cd ad  52
3b19 : bd 02 cd b7 02 d0 c5 a9  53
3b21 : 00 8d bc 02 8d bd 02 ee  04
3b29 : bb 02 d0 03 ee ba 02 ad  a2
3b31 : ba 02 cd b4 02 d0 ad ad  af
3b39 : bb 02 cd b5 02 d0 a5 60  1d
3b41 : ad b0 02 29 7f 20 d2 ff  30
3b49 : ad b1 02 29 7f 20 d2 ff  b9
3b51 : ad b2 02 f0 06 18 69 23  03
3b59 : 20 d2 ff a9 28 20 d2 ff  e6
3b61 : ad b3 02 0a a8 b9 b9 02  ed
3b69 : aa b9 b8 02 20 cd bd ac  1f
3b71 : b3 02 88 f0 2b a9 2c 20  56
3b79 : d2 ff 98 0a a8 b9 b9 02  f5
3b81 : aa b9 b8 02 20 cd bd ac  37
3b89 : b3 02 88 88 f0 12 a9 2c  0f
3b91 : 20 d2 ff 98 0a a8 b9 b9  6d
3b99 : 02 aa b9 b8 02 20 cd bd  4a
3ba1 : a9 29 20 d2 ff a9 3d 20  c4
3ba9 : d2 ff ad b2 02 d0 1d a5  a3
3bb1 : fb a4 fc 20 a2 bb 20 d7  7a
3bb9 : bd a9 0d 20 d2 ff a5 fb  4e
3bc1 : 18 69 05 85 fb a5 fc 69  33
3bc9 : 00 85 fc 60 c9 01 d0 3d  39
3bd1 : a0 00 b1 fb 85 02 c8 b1  4c
3bd9 : fb 85 fd c8 b1 fb 85 fe  3e
3be1 : a9 22 20 d2 ff a5 02 f0  15
3be9 : 0c a0 00 b1 fd 20 d2 ff  a8
3bf1 : c8 c4 02 90 f6 a9 22 20  33
3bf9 : d2 ff a9 0d 20 d2 ff a5  bb
3c01 : fb 18 69 03 85 fb a5 fc  8c
3c09 : 69 00 85 fc 60 a0 00 b1  e2
3c11 : fb aa c8 b1 fb a8 8a 20  39
3c19 : 95 b3 20 dd bd 20 1e ab  f8
3c21 : a9 0d 20 d2 ff a5 fb 18  00
3c29 : 69 02 85 fb a5 fc 69 00  5c
3c31 : 85 fc 60 20 de 78 6c 02  b8
3c39 : a0 a0 01 84 0f a5 39 85  08
3c41 : 14 a5 3a 85 15 20 13 a6  53
3c49 : a0 03 20 2e 7c c8 b1 5f  cc
3c51 : d0 01 60 c9 80 b0 10 20  01
3c59 : d2 ff c9 22 d0 ec a5 0f  0b
3c61 : 49 ff 85 0f 4c 4b 7b c9  8e
3c69 : ff f0 ec 24 0f 30 e8 c9  4a
3c71 : cc f0 2b c9 cd f0 45 84  3c
3c79 : 49 38 e9 7f aa 84 49 a0  7e
3c81 : ff ca f0 08 c8 b9 9e a0  39
3c89 : 10 fa 30 f5 c8 b9 9e a0  f7
3c91 : 30 05 20 d2 ff d0 f5 a4  4e
3c99 : 49 29 7f 4c 58 7b c8 84  6e
3ca1 : 49 b1 5f aa a0 ff ca f0  07
3ca9 : 08 c8 b9 3c 8a 10 fa 30  81
3cb1 : f5 c8 b9 3c 8a 30 e0 20  ee
3cb9 : d2 ff d0 f5 c8 84 49 b1  b7
3cc1 : 5f aa a0 ff ca f0 08 c8  83
3cc9 : b9 3c 8b 10 fa 30 f5 c8  20
3cd1 : b9 3c 8b 30 c2 20 d2 ff  0a
3cd9 : d0 f5 a9 00 8d d0 02 60  36
3ce1 : 38 20 f0 ff 8a 48 98 48  43
3ce9 : a2 00 20 ff e9 a2 01 20  8b
3cf1 : ff e9 a2 02 20 ff e9 a2  bd
3cf9 : 00 a0 00 18 20 f0 ff a6  23
```

Listing des Sprite+Grafik-Basic. Bitte mit dem MSE eingeben.

```
3d01 : 39 a5 3a 20 cd bd a9 20  51
3d09 : 20 d2 ff 20 3a 7b 68 a8  09
3d11 : 68 aa 18 4c f0 ff a9 92  39
3d19 : 20 d2 ff a5 9d d0 0b 20  24
3d21 : e1 7b a5 c6 f0 fc a9 00  a0
3d29 : 85 c6 4c 64 89 84 02 a5  c1
3d31 : 5f 18 65 02 85 45 a5 60  10
3d39 : 69 00 85 62 a5 45 c5 7a  e1
3d41 : d0 13 a5 62 c5 7b d0 0d  e6
3d49 : a9 12 20 d2 ff a4 02 a9  de
3d51 : 01 8d e0 02 60 ad e0 02  8c
3d59 : d0 03 a4 02 60 a9 00 8d  83
3d61 : e0 02 a9 92 20 d2 ff a4  e1
3d69 : 02 60 78 a9 00 8d 12 d0  45
3d71 : ad 11 d0 29 7f 8d 11 d0  4a
3d79 : a9 81 8d 1a d0 a9 8a 8d  29
3d81 : 14 03 a0 7c 8c 15 03 58  fd
3d89 : 60 ad 19 d0 8d 19 d0 29  57
3d91 : 80 d0 03 4c 31 ea ad 12  09
3d99 : d0 c9 49 b0 1a a0 49 8c  9b
3da1 : 12 d0 a9 c8 8d 16 d0 a9  bf
3da9 : 1b 8d 11 d0 a9 05 8d 18  12
3db1 : d0 68 a8 68 aa 68 40 a0  1d
3db9 : 00 8c 12 d0 ad a1 02 f0  70
3dc1 : f0 a9 09 8d 18 d0 a9 3b  9f
3dc9 : 8d 11 d0 ad a2 02 f0 e1  8a
3dd1 : a9 d8 8d 16 d0 4c b2 7c  40
3dd9 : a9 7c 8d 09 03 a9 17 8d  3a
3de1 : 08 03 4c 6b 7c 78 a9 89  31
3de9 : 8d 09 03 a9 64 8d 08 03  ca
3df1 : ea a9 31 8d 14 03 a9 ea  84
3df9 : 8d 15 03 58 60 a9 03 20  7c
3e01 : fb a3 20 06 a9 18 98 65  1f
3e09 : 7a 48 a5 7b 69 00 48 a5  83
3e11 : 39 48 a5 3a 48 a9 52 48  cb
3e19 : 4c ae a7 20 79 00 d0 03  8b
3e21 : 4c 08 af 68 68 20 8a ad  77
3e29 : a8 ba bd 01 01 c9 52 f0  47
3e31 : 05 a2 1f 4c 37 a4 98 d0  75
3e39 : 17 bd 02 01 85 3a bd 03  f7
3e41 : 01 85 39 bd 04 01 85 7b  60
3e49 : bd 05 01 85 7a 4c ae a7  8e
3e51 : 8a 18 69 07 aa 9a 4c ae  31
3e59 : a7 20 9e ad 20 79 00 c9  cf
3e61 : 89 f0 05 a9 a7 20 ff ae  b2
3e69 : a5 61 8d e1 02 f0 03 4c  ab
3e71 : 40 a9 a0 ff c8 b1 7a d0  53
3e79 : 03 4c 3b a9 c9 cd d0 f4  df
3e81 : c8 b1 7a c9 27 d0 ed 20  eb
3e89 : f8 a8 20 73 00 c9 cd d0  73
3e91 : f6 20 73 00 c9 27 d0 ef  6d
3e99 : 20 73 00 4c 40 a9 ad e1  c8
3ea1 : 02 4c 6e 7d a0 ff c8 b1  a5
3ea9 : 7a 99 00 02 f0 04 c9 3a  fb
3eb1 : d0 f4 a9 01 85 7a a9 04  61
3eb9 : 85 7b a0 00 b1 7a f0 0d  f1
3ec1 : c9 cd f0 1d e6 7a d0 02  da
3ec9 : e6 7b 4c bd 7d c8 b1 7a  11
3ed1 : d0 0a c8 b1 7a d0 05 a2  96
3ed9 : 20 4c 37 a4 a0 00 4c c5  48
3ee1 : 7d c8 b1 7a c9 2b d0 f4  a1
3ee9 : c8 b9 fe 01 f0 0b c9 3a  71
3ef1 : f0 07 d1 7a f0 f2 4c dd  bc
3ef9 : 7d 4c f8 a8 e0 80 90 03  4a
3f01 : 4c 74 a4 8a 48 a5 9d d0  cb
3f09 : 2d a2 01 c6 7a a5 7a c9  f3
3f11 : ff d0 02 c6 7b ca 10 f3  08
3f19 : a9 05 8d 18 d0 a9 1b 8d  8d
3f21 : 11 d0 a9 c8 8d 16 d0 a6  38
3f29 : 39 a5 3a 20 cd bd 20 3a  87
3f31 : 7b a9 0d 20 d2 ff a5 39  fe
3f39 : 8d e2 02 a5 3a 8d e3 02  10
3f41 : 68 8d e4 02 10 03 4c 74  1c
3f49 : a4 c9 1f b0 03 4c 3b a4  79
3f51 : 38 e9 1f 0a aa bd 70 7e  de
3f59 : 85 22 bd 71 7e 85 23 20  6e
3f61 : cc ff a9 00 85 13 20 d7  b8
3f69 : aa 20 45 ab 4c 54 a4 80  e5
3f71 : 7e 94 7e aa 7e b8 7e cd  71
3f79 : 7e 98 7f ff 00 00 ff 55  ce
3f81 : 4e 54 49 4c 20 57 49 54  60
3f89 : 48 4f 55 54 20 52 45 50  a3
3f91 : 45 41 d4 4c 41 42 45 4c  09
3f99 : 20 4e 4f 54 20 46 4f 55  5b
3fa1 : 4e c4 49 4c 4c 45 47 41  bc
3fa9 : 4c 20 53 50 52 49 54 45  30
3fb1 : 20 4e 55 4d 42 45 d2 49  24
3fb9 : 4c 4c 45 47 41 4c 20 53  03
3fc1 : 50 52 49 54 45 20 53 54  63
3fc9 : 52 49 4e c7 53 50 52 49  e0
3fd1 : 54 45 20 44 45 53 49 47  fb
3fd9 : 4e 49 4e 47 20 43 4f 4f  40
3fe1 : 52 44 49 4e 41 54 c5 20  80
3fe9 : 20 20 20 20 20 20 20 20  e9
3ff1 : 00 00 ad e4 02 10 01 60  5f
3ff9 : c9 1f b0 2a 0a aa bd 26  fc
4001 : a3 85 22 bd 27 a3 85 23  93
4009 : 20 d7 aa 20 45 ab a0 00  f8
4011 : b1 22 48 29 7f 20 47 ab  78
4019 : c8 68 10 f4 a9 69 a0 a3  68
4021 : 20 1e ab 4c 4d 7f 38 e9  4a
4029 : 1f 0a aa bd 70 7e 85 22  05
4031 : bd 71 7e 85 23 4c 09 7f  af
4039 : ad e3 02 c9 ff d0 01 60  dd
4041 : ae e2 02 ad e3 02 20 cd  01
4049 : bd 4c 3a 7b ad e2 02 85  2f
4051 : 14 ad e3 02 85 15 20 13  1d
4059 : a6 4c c9 a6 a9 03 20 fb  98
4061 : a3 a5 7b 48 a5 7a 48 a5  59
4069 : 3a 48 a5 39 48 a9 33 48  87
4071 : 20 a5 7d 4c ae a7 a9 ff  1b
4079 : 85 4a 20 8a a3 9a c9 33  19
4081 : f0 05 a2 24 4c 37 a4 68  03
4089 : 68 85 39 68 85 3a 68 85  e6
4091 : 7a 68 85 7b 4c f8 a8 53  e6
4099 : 55 42 45 4e 44 20 57 49  60
40a1 : 54 48 4f 55 54 20 43 41  6e
40a9 : 4c cc ff 8a 48 ad a1 02  29
40b1 : f0 08 20 92 c5 a9 00 8d  c4
40b9 : a1 02 68 aa 4c fd 7d 00  75
40c1 : 00 ff ff 00 00 ff f7 00  a1
40c9 : 00 ff ff 00 00 ff f7 00  a9
40d1 : 04 ff ff 00 00 ff f7 00  b5
40d9 : 00 ff ff 00 00 ff f7 00  b9
40e1 : 00 ff ff 00 00 ff f7 00  c1
40e9 : 00 ff ff 00 00 ff f7 00  c9
40f1 : 00 ff ff 00 00 ff f7 00  d1
40f9 : 00 ff ff 00 00 ff ff ff  f9
```

Listing des Sprite+Grafik-Basic (Schluß). Mit dem MSE eingeben.



Hardcopy CP-80X mit Simons Basic

Mit diesem Programm können Sie Hardcopies von HiRes-Grafiken in Simons Basic erstellen.

Das Programm erstellt eine Hardcopy der HiRes-Grafik von Simons Basic. Dem CP-80X ist zwar ein Heft mit mehreren Routinen für den C 64 beigelegt, doch die darin aufgeführte Hardcopy für Simons Basic funktionierte leider nicht. Da der Drucker noch relativ neu auf dem Markt und nicht ganz Epson-kompatibel ist (wegen des eingebauten Interfaces), ist bisher noch keine Hardcopy erschienen. Dieses Programm schafft »schnelle« Abhilfe (Maschinensprache, Druckzeit: 58 Sekunden).

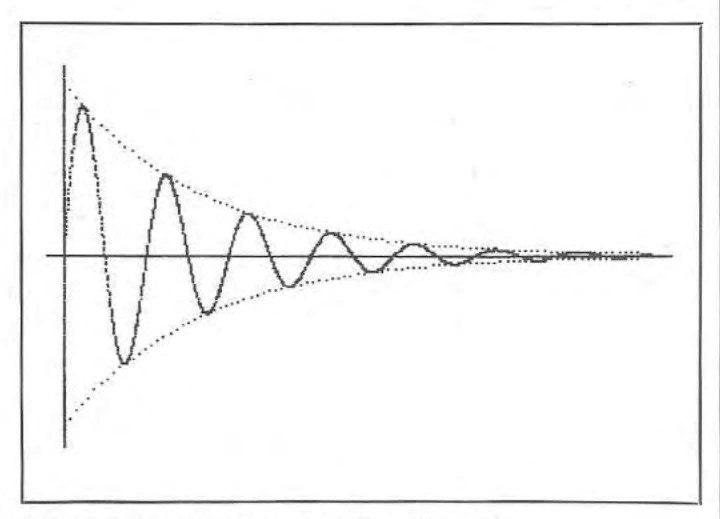
Nach dem Abtippen des Programms (Abspeichern nicht vergessen!) wird es mit RUN gestartet.

Es liest dann die Daten ein und überprüft diese auf ihre Richtigkeit. Da in jeder Zeile Prüfsummen eingebaut sind, können eventuelle Fehler auf eine Zeile genau lokalisiert werden.

Danach wird das Maschinenspracheprogramm gestartet, dieses löscht das Ladeprogramm und schützt das Maschinenprogramm vor dem Überschreiben durch Basic.

Dann kann durch »SYS 32000« jederzeit eine Hardcopy der Grafikseite erstellt werden. Die Hardcopy wird in der Mitte des Blattes ausgedruckt, nach dem Drucken wird der Zeilenabstand zurück auf 1/6 inch gestellt.

(Stephan Arndt/rg)



Beispiel einer Hardcopy auf dem CP-80X

```
100 REM ********************************          <238>
101 REM * HARDCOPY MIT SIMON'S BASIC   *          <222>
103 REM *          AUF CP-80X          *          <187>
104 REM *                              *          <153>
105 REM * 1-1985 BY  STEPHAN ARNDT     *          <165>
106 REM *            SUEDWESTKORSO 62 *           <201>
107 REM *            1000 BERLIN 41    *          <218>
108 REM *                              *          <157>
109 REM ********************************          <247>
110 :                                             <086>
120 : REM HARDCOPY DER HIRES - SEITE              <121>
130 : REM  MIT ' SYS 32000 '                      <234>
140 :                                             <116>
150 FOR I = 32000 TO 32293 STEP 8                 <063>
160 S = 0                                         <133>
170 FOR K = 0 TO 7                                <255>
180 READ A : S = S + A                            <106>
190 POKE I+K , A                                  <036>
200 NEXT K                                        <044>
210 READ R : IF S = R THEN 230                    <200>
220 PRINT " FEHLER IN ";300+N*10 :END             <142>
230 N=N+1:NEXT I                                  <106>
240 :                                             <216>
250 SYS 32284                                     <158>
260 :                                             <238>
300 DATA  76, 64,125,  0,224, 64,225,128,
    906                                           <093>
310 DATA 226,192,227,  0,229, 64,230,128,1
    296                                           <153>
320 DATA 231,192,232,  0,234, 64,235,128,1
    316                                           <085>
330 DATA 236,192,237,  0,239, 64,240,128,1
    336                                           <189>
340 DATA 241,192,242,  0,244, 64,245,128,1
    356                                           <159>
350 DATA 246,192,247,  0,249, 64,250,128,1
    376                                           <007>
360 DATA 251,192,252,  0,254,  0,  0,  0,
    949                                           <030>
370 DATA   0,  0,  0,  0,  0, 25,  8, 40,
     73                                           <107>
380 DATA 234,169,  4,133,186,169,'96,133,1
    124                                           <029>
390 DATA 185,169,  0,133,183,169,  1,133,
    973                                           <205>
400 DATA 184, 32,192,255,166,184, 32,201,1
    246                                           <252>
410 DATA 255,169, 27, 32,210,255,169, 51,1
    168                                           <134>
420 DATA  32,210,255,169, 22, 32,210,255,1
    185                                           <002>
430 DATA 162,  0,142, 61,125,174, 61,125,8
    50                                            <040>
440 DATA 138, 10,170,189,  3,125,133,251,1
    019                                           <170>
450 DATA 189,  4,125,133,252,160, 20,169,1
    052                                           <220>
460 DATA  32, 32,210,255,136,208,250,169,1
    292                                           <057>
470 DATA  27, 32,210,255,169, 76, 32,210,1
    011                                           <124>
480 DATA 255,169, 64, 32,210,255,169,  1,1
    155                                           <192>
490 DATA  32,210,255,162,  0,142, 63,125,
    989                                           <108>
500 DATA 160,  0,140, 62,125,120,120,165,
    892                                           <203>
510 DATA   1, 41,253,133,  1, 41,253,234,
    957                                           <003>
520 DATA 172, 62,125,177,251,162,  0,234,1
    183                                           <130>
530 DATA  10, 62, 53,125,232,224,  8,208,
    922                                           <033>
540 DATA 246,172, 62,125,200,140, 62,125,1
    132                                           <069>
550 DATA 192,  8,208,227,165,  1,  9,  2,
    812                                           <249>
560 DATA 133,  1, 88,160,  0,185, 53,125,
    745                                           <248>
570 DATA  32,210,255,200,192,  8,208,245,1
    350                                           <210>
580 DATA 234, 24,165,251,105,  8,133,251,1
    171                                           <116>
590 DATA 165,252,105,  0,133,252,174, 63,1
    144                                           <142>
600 DATA 125,232,142, 63,125,224, 40,208,1
    159                                           <087>
610 DATA 167,169, 13, 32,210,255,174, 61,1
    081                                           <026>
620 DATA 125,232,142, 61,125,224, 25,240,1
    174                                           <070>
630 DATA   3, 76,109,125,234,169, 27, 32,
    775                                           <050>
640 DATA 210,255,169, 50, 32,210,255, 32,1
    213                                           <129>
650 DATA 204,255, 88, 96,169,125,141,155,1
    233                                           <112>
660 DATA 129, 76, 10,128,  0,  0,  0,  0,3
    43                                            <037>
670 :                                             <138>
```

© 64'er **Listing »Hardcopy CP-80X«. Bitte beachten Sie bei der Eingabe den Checksummer.**

»Multicolor« auf dem FX/RX-80

Hardcopy-Programme für den Epson FX gibt es viele. Doch dieses Programm ermöglicht eine Umsetzung der Farben in verschiedene Graustufen.

Die meisten Drucker-Besitzer werden sich schon geärgert haben, daß »normale« Hardcopies von Multicolor-Grafiken kaum als solche zu erkennen waren. Besitzer eines Epson-RX/FX-80 sind jetzt fein raus.

Ich habe mich geärgert, als ich das erste Mal die Hardcopy einer Strip-Poker-Grafik auf meinem FX-80 ausdrucken wollte. Als ich dann noch den Test des PRINT-64-Interfaces in der 64'er, Ausgabe 12/84, gelesen habe, das den Ausdruck von Multicolor-Grafiken mittels Grauschattierungen erlaubt, habe ich mir gedacht: Das muß doch auch softwaremäßig machbar sein.

Gesagt, getan. Eines Abends nahm ich mir eine Flasche Cola und etwas Knabberkram und fing an zu programmieren. Nach mehreren erfolglosen Anläufen und stapelweise bedrucktem Papier war ich am Ziel: »M-C-HRDCPY« lief. Doch nun zum eigentlichen Programm.

Das Prinzip

Eine Multicolor-Grafik ist ja im wesentlichen genauso wie eine normale Grafik aufgebaut, nur repräsentieren jetzt zwei statt einem Bit einen Punkt auf dem Bildschirm. In dieser Kombination aus den beiden Bits steckt jetzt noch eine Farbinformation: Sind beide Bits 0, so wird der Punkt in der Farbe des Hintergrundregisters gesetzt, sind sie jedoch 01, 10 oder 11, wird der Punkt in der Farbe aus den Registern 33 bis 36 des VIC gesetzt.

Ich verfahre nun so, daß ich jeden Bildschirmpunkt in einer Matrix von 2x12 Punkten auf dem Drucker darstelle. Durch die Anordnung der Punkte in dieser Matrix kann ich nun verschiedene Graustufen darstellen — die Grundlage ist damit gegeben.

Die Bedienung

Die Bedienung mag auf den ersten Blick etwas kompliziert erscheinen, erlaubt dafür aber auch optimale Variationsmöglichkeiten.

Vor jeder Anwendung kommt aber leider noch das Abtippen von 438 DATAs. Wenn Sie das gemacht haben, sollten Sie das Programm vorsichtshalber vor dem Starten speichern (so mancher mußte wegen eines Tippfehlers noch einmal anfangen ...). Nach dem Starten sollte sich das Programm nach zirka vier Sekunden mit der Frage »Startadresse?« melden. Geben Sie nun ein, wo Sie das Programm im Speicher haben wollen. Eine kleine Hilfe hierzu gibt die Tabelle, die günstige Startadressen für verschiedene Programme nennt.

Dann wird die Floppy anlaufen (falls Sie nur eine Datasette besitzen — bei Vorhandensein eines Epson-Druckers recht unwahrscheinlich — müssen Sie die Zeilen 1250 bis 1280 herausnehmen und zum Arbeiten immer wieder den Basic-Lader benutzen) und ein File namens »M-C-HRDCPY (Startadresse)« wird auf Diskette erzeugt. Gleichzeitig wird das Programm adressenangepaßt in den Speicher ab (Adresse) geschrieben. Wenn Sie »M-C-HRDCPY« nun noch eventuell durch POKE 55,(adresse) INT(adresse/256)*256:POKE 56,INT(adresse/256): CLR schützen, kann der erfolgreichen Hardcopy eigentlich nur noch das Dreckfuhlerteufelchen im Wege stehen.

Nun zum Aufruf der Hardcopy-Routine. Generell hat er die Form:

SYS (adresse),(filenummer),(page),a,b,c,d

(adresse) ist die Startadresse des Programms — die, die Sie auf die Frage »Startadresse?« eingegeben haben —, für

(filenummer) setzen Sie eine 0 ein, falls Sie Ihren Drucker am User-Port angeschlossen haben oder die Filenummer eines Files, das Sie vorher auf den Drucker zur Übergabe von Grafikdaten geöffnet haben.

(page) ist die Speicherseite, auf der die Grafik beginnt — also INT(Adresse der Grafik/256) und

a,b,c,d sind die Masken für die Graustufen der Hardcopy.

Eine Hardcopy, bei der die Farben durch verschiedene Graustufen dargestellt werden.

Wie bereits erwähnt, kennt der VIC vier Möglichkeiten für die Darstellung eines Punktes, nämlich die Kombinationen 00, 01, 10 oder 11. Andererseits kennt die Hardcopy-Routine vier Graustufen. Die Parameter a, b, c und d geben nun an, wie die Bitkombinationen den Graustufen zugeordnet werden. a gibt an, welche Bitkombination auf dem Drucker ganz hell, b, welche hellgrau, c, welche dunkelgrau und d, welche schwarz dargestellt werden. a, b, c und d müssen dabei den dezimalen Wert der Bitkombinationen annehmen, das heißt, 0 für 00, 1 für 01, 2 für 10 und 3 für 11. Ein Beispiel hierzu gibt der Hardcopy-Ausdruck.

Hier noch einmal zur Übung:

Angenommen, Sie haben eine Simons Basic-Multicolor-Grafik und wollen diese so ausdrucken, daß die 00-Punkte schwarz, die 01-Punkte hellgrau, die 10-Punkte weiß und die 11-Punkte dunkelgrau dargestellt werden.

Also laden Sie den Basic-Lader, sehen in der Tabelle nach, welche Startadresse günstig ist, geben diese ein und tippen dann, nachdem sich der Computer wieder mit READY gemeldet hat, ein:

POKE 55,44:POKE 56,126:CLR

um M-C-HRDCPY vor dem Überschreiben zu schützen und dann weiter:

SYS 32300,0,224,2,1,3,0

falls Sie den Drucker am User-Port betreiben beziehungsweise

OPEN xx,(Druckeradresse),(Grafiksekundäradresse)
SYS 32300,xx,224,2,1,3,0

falls Sie ein Hardware-Interface besitzen.

Tabelle. Günstige Startadresse und Werte für (page) für verschiedene Programme und Werte für POKE 55,x:POKE 56,y:CLR

Programm	Startadresse	page	poke 55,	poke 56,
Dia-Show	49152	32	-	-
Simons Basic	32300	224	44	126
Paint Magic (Grafik laden + starten)	49152	64	-	-
Supergrafik 64	26666	160+224	42	104
Screen Graphics 64	32300	160	44	126

Das Programm wurde auf einem FX-80 sowohl mit Software- als auch mit Data-Becker- und Görlitz-Hardwareinterface und auf einem RX-80 mit Software- und Görlitz-Hardwareinterface getestet und lief in allen Fällen problemlos. Vermutlich ist das Programm auch auch dem MX-80 und anderen Epson- und Epson-kompatiblen Druckern lauffähig. Im Zweifelsfall müssen Sie daher ausprobieren, ob auch Ihr Drucker für dieses Programm geeignet ist, kaputt kann eigentlich nichts gehen. Benutzt werden folgende Steuersequenzen: ESC 1, um den Drucker auf 7/72 Inch Zeilenvorschub zu schalten, ESC (Klammeraffe), um ihn zu initialisieren und ESC Z, um ihn in 1920 Punkte/Zeile-Grafikmodus zu bringen.

Die Kompatibilität

Da das Programm also recht viele Drucker unterstützt, frei verschiebbar ist, den Ausdruck von Grafiken aus den versteckten RAMs erlaubt und sowohl Hard- als auch Software-Interfaces unterstützt, sollte meiner Meinung nach ein Höchstmaß an Kompatibilität gewährleistet sein.

Ansonsten viel Spaß und gut Druck!

(Karsten Klenz/rg)

```
20 REM================================      <039>
30 REM=      MULTI-COLOR-HARDCOPY      =    <080>
40 REM=                                =    <034>
50 REM= FUER C-64 UND EPSON FX/RX-80   =    <182>
60 REM=                                =    <054>
70 REM=KARSTEN KLENZ,HARBURGER STR. 24=     <032>
80 REM=2105 SEEVETAL 1,TEL.:04105/2695=     <160>
90 REM================================      <109>
1000 DIM A%(422)                            <019>
1010                                        <248>
1020 FOR T=1 TO 422                         <212>
1030   READ A%(T)                           <054>
1040   S=S+A%(T)                            <218>
1050 NEXT T                                 <206>
1060                                        <044>
1070 IF S<>43018 THEN PRINT"FEHLER IN DATA
     S !":STOP                              <034>
1080                                        <064>
1090 INPUT"STARTADRESSE ";AD                <021>
1100 FOR T=1 TO 16                          <068>
1110   READ B                               <142>
1120   A=A%(B+2)+A%(B+3)*256+AD             <092>
1130   A%(B+2)=A-INT(A/256)*256:A%(B+3)=I
     NT(A/256)                              <186>
1140 NEXT T                                 <040>
1200                                        <184>
1210 FOR T=1 TO 422                         <148>
1220   POKE AD+T-1,A%(T)                    <196>
1230 NEXT T                                 <130>
1240                                        <224>
1250 OPEN 1,8,14,"@:M-C-HRDCPY"+STR$(AD)+"
     ,P,W"                                  <209>
1260 PRINT#1,CHR$(AD-INT(AD/256)*256)CHR$(
     INT(AD/256));                          <056>
1270 FOR T=1 TO 422:PRINT#1,CHR$(A%(T));:N
     EXT T                                  <178>
1280 CLOSE 1                                <021>
1290 END                                    <022>
1300 REM= PROGRAMM-DATAS ===============    <119>
3681 DATA 32,253,174,32,158,183,134,2,240,
     5,32,201,255,208,0,32,253,174,32       <128>
3682 DATA 158,183,134,21,169,0,133,20,32,2
     53,174,32,158,183,134,87,32,253,174    <123>
3683 DATA 32,158,183,134,88,32,253,174,32,
     158,183,134,89,32,253,174,32,158       <242>
3684 DATA 183,134,90,169,255,141,3,221,173
     ,2,221,9,4,141,2,221,160,1,185,114     <212>
3685 DATA 1,32,79,1,136,16,247,169,0,133,9
     8,169,0,133,99,32,228,255,240,3,76     <177>
3686 DATA 59,1,160,3,185,110,1,32,79,1,136
     ,16,247,169,0,133,97,165,20,164,21     <115>
3687 DATA 133,34,132,35,164,99,162,0,120,1
     69,53,133,1,177,34,149,100,200,232     <248>
3688 DATA 224,4,208,246,169,55,133,1,88,16
     9,3,133,96,162,3,181,100,164,96,240    <029>
3689 DATA 6,74,74,136,76,156,0,24,41,3,149
     ,91,202,16,236,160,0,162,3,169,0       <104>
3690 DATA 133,95,169,3,133,104,181,91,197,
     87,208,6,185,118,1,24,144,23,197       <239>
3691 DATA 88,208,6,185,130,1,24,144,13,197
     ,89,208,6,185,142,1,24,144,3,185       <081>
3692 DATA 154,1,37,104,5,95,133,95,24,38,1
     04,24,38,104,24,202,16,205,32,79       <088>
3693 DATA 1,200,192,12,208,187,166,96,202,
     134,96,16,156,24,165,34,105,8,133      <016>
3694 DATA 34,165,35,105,0,133,35,166,97,23
     2,134,97,224,40,240,3,76,122,0,166     <215>
3695 DATA 99,232,232,232,232,134,99,224,8,
     240,3,76,91,0,24,165,20,105,64,133     <226>
3696 DATA 20,165,21,105,1,133,21,166,98,23
     2,134,98,224,25,240,3,76,87,0,160      <144>
3697 DATA 1,185,116,1,32,79,1,136,16,247,3
     2,204,255,165,2,32,195,255,96,166      <088>
3698 DATA 2,240,3,76,210,255,141,1,221,169
     ,16,44,13,221,240,251,173,0,221        <087>
3699 DATA 9,4,141,0,221,41,251,141,0,221,9
     6,7,128,90,27,49,27,64,27,0,0,0,0      <003>
3700 DATA 0,0,0,0,0,0,0,0,170,0,85,0,170,0
     ,85,0,170,0,85,0,170,0,0,0,85,0,0      <129>
3701 DATA 0,170,0,0,0,170,85,170,85,170,85
     ,170,85,170,85,170,85                  <091>
5000 REM= RELATIVE ADRESSEN ===========     <212>
5010 DATA 74,77,96,101,104,161,190,200,210
     ,216,235,272,287,312,317,320           <087>
```

© 64'er Listing »M-C-Hardcopy«.
Bitte den Checksummer 64 beachten.

Große Hardcopy auf Star Gemini

Posterähnliche Hardcopies können die Star Gemini-Besitzer mit diesem Programm erzeugen.

Geben Sie das Programm genau nach Programmlisting, von Basic aus ein, vergleichen Sie es und speichern Sie es ab. Danach können Sie den Basic-Lader mit RUN starten. Das Programm sollte sich mit »Datas OK !«, »AUFRUF ; SYS 49152,ADR,FN,GN,SA,INV« melden und ist nun ab sofort startbereit.

Sollte diese Meldung nicht erscheinen und »Fehler in Datas !!!« gemeldet werden, so ist eine Diskrepanz in der Prüfsumme aufgetreten. Die DATA-Zeilen sind fehlerhaft. Kontrollieren Sie das Programm noch einmal genau anhand des Listings und starten Sie es erneut.

Ist das Programm bereit, so können Sie den Basic-Lader mit NEW aus dem Arbeitsspeicher löschen. Laden Sie nun eine

> **Achdung - Kombjuder-Raum !**
>
> Der Raum is vull bis under die Deggn mit di dollsdn und deiersdn elegdrischn und vullelegdronischn Abbarade.
>
> Stauna und glodzn derf dou jeder, obber umananderworschdln und an die Gnöpfler rumdriggen, des derfen blouß mir, die Eggsberden !

Beispiel Hardcopy (das Original ist A4 groß)

bereits abgespeicherte Grafik, oder erzeugen Sie per Programm den gewünschten HiRes-Bildschirm. Zum Poster-Ausdruck starten Sie die Routine mit »SYS 49152, ADR, FN, GN, SA, INV«. ADR gibt dabei die Startadresse des 8-KByte-Bereiches an. Oft ist dies die Adresse 8192 ($2000), es ist jedoch auch möglich, Bildschirme, die unter dem ROM stehen, auszudrucken. Für Simons Basic wäre dies zum Beispiel die Adresse 57344 ($D000). Die drei nächsten Parameter entsprechen den Angaben des OPEN-Befehls. Sie beinhalten die Filenummer, eine Zahl zwischen 1 und 255, die für den Ausdruck belanglos ist, die Geräteadresse des Druckers, die im allgemeinen auf 4 eingestellt ist und die Sekundäradresse für die Drucker- beziehungsweise Interface-Ansteuerung. Für das Universal-Interface (WW Interface 92000/G, C 64/Centr.) muß diese gleich eins sein. Der letzte Parameter gibt den Druckmodus an. Eine »0« steht für normale Darstellung; eine »1« für inverse.

Das Poster kann auf beliebiges Papier gedruckt werden. Für die Verwendung von Einzelblättern wird allerdings die Papierende-Kennung ausgeschaltet, damit die gesamte Hardcopy ohne Unterbrechung ausgedruckt werden kann. Nach dem Drucken wird durch ein Form-Feed das Papier aus dem Drucker geschoben, der Drucker in den Einschaltzustand versetzt und das entsprechende File geschlossen. Wird die Hardcopy mit der Stop-Taste abgebrochen, so wird ebenso ein Druckerreset ausgeführt und das Druckerfile geschlossen. Das Papier wird allerdings nicht transportiert. Der Computer sollte sich mit »BREAK«, »READY« wieder melden.

Mit Hilfe eines Monitors oder eines modifizierten Laders kann die Hardcopy auch direkt abgespeichert werden (Achtung: Der Monitor darf nicht bei 49152 ($C000) liegen).
Die Adressen, die das Programm belegt, reichen von 49152 bis 49590.

Natürlich kann die Routine in eigene Grafikprogramme eingebettet werden. Die Filenummer sollte dann einen noch unbelegten Kanal verwenden.

Arbeitet man mit Simons Basic zusammen, so ist folgendes zu beachten : Simons Basic belegt den Speicherbereich von $C000 bis $C400 mit der Farbinformation der Grafik. Das heißt, unsere Routine würde durch den HiRes-Befehl gelöscht. Um trotzdem mit der Erweiterung arbeiten zu können, ist die Grafik nach Beendigung der Bilderstellung mit »CSET 0« auszuschalten. Jetzt erst können wir unsere Routine mit dem Basic-Lader nach 49152 ($C000) kopieren und starten. Achtung: Durch das Kopieren und Starten der Routine geht die Farbinformation der ersten Bildhälfte verloren. Will man die Farbe wieder setzen, so muß der veränderte Bereich nach der Hardcopy wieder mit der Farbinformation gefüllt werden (FOR i=49152 TO 49600:POKE i,PEEK(49700):NEXT i).

Die Arbeit mit dem »64'er«-Programm »Hi-Eddi« ist am besten möglich, indem man die Grafikseite abspeichert und nach Verlassen des Programmes wieder lädt. Die Zusammenarbeit mit anderen professionellen Programmen sollten Sie selbst erproben.

Abschließend noch ein Wort zur Anpassung an andere Drucker. Der Drucker muß auf alle Fälle Epson- beziehungsweise Star-kompatibel sein und Blockgrafikcharakter besitzen. (Für Profis: die acht Charakter beginnen in der vorletzten Zeile, mit der vierten Zahl von hinten.)

Weiterhin ist zu beachten, daß die Sekundäradresse so gewählt werden muß, daß alle Grafikbefehle angenommen werden (beim Universal-Interface zum Beispiel 1).

Das ist doch eine Hardcopy, die wirklich Format zeigt!

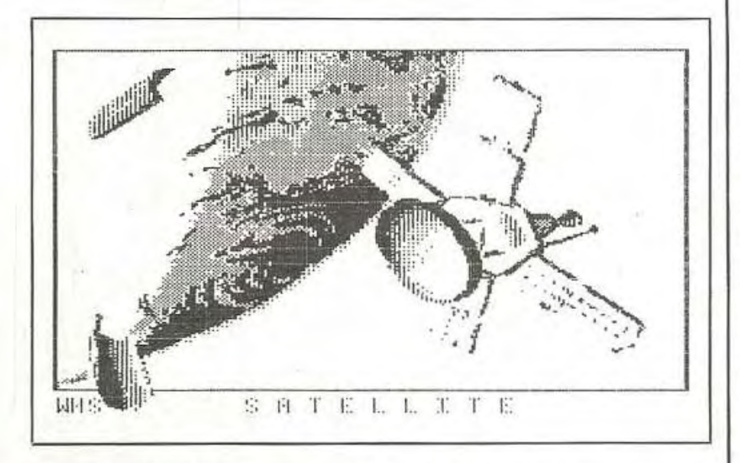
(Thomas Hohenberger/rg)

```
100 REM" ↑****************************                <123>
110 REM" ={25SPACE}=                                  <085>
120 REM" ={4SPACE}SUPER - POSTER 64{4SPACE
         }=                                           <059>
130 REM" ={25SPACE}=                                  <105>
140 REM" ={5SPACE}STAR GEMINI 10X{5SPACE}=            <212>
150 REM" ={2SPACE}& UNIVERSAL-INTERFACE{2S
         PACE}=                                       <226>
160 REM" ={25SPACE}=                                  <135>
170 REM" ={6SPACE}BY T. HUHENBERGER{2SPACE
         }=                                           <142>
180 REM" ={25SPACE}=                                  <155>
190 REM" 7****************************Y               <104>
200 :                                                 <176>
210 RESTORE                                           <004>
220 SU = 0                                            <004>
230 :                                                 <206>
240 FOR AD = 49152 TO 49567                           <212>
250 : READ P%                                         <199>
260 : POKE AD,P%                                      <158>
270 : SU = SU + P%                                    <153>
280 NEXT AD                                           <114>
```

Listing »Superposter 64«

```
290 :                                              <012>
300 IF SU <> 57112 THEN PRINT "FEHLER IN D
    ATAS !!!" : STOP                               <178>
310 :                                              <032>
320 PRINT "DATAS OK !"                             <035>
330 PRINT                                          <178>
340 PRINT "{DOWN}AUFRUF : SYS49152,ADR,FN,
    GN,SA,INV"                                     <242>
350 :                                              <072>
360 END                                            <108>
370 :                                              <092>
380 :                                              <102>
10000 DATA 32,253,174,32,138,173,32,247,18
      3,140,126,193,141,127,193,32,241,183 <202>
10010 DATA 142,128,193,32,241,183,142,129,
      193,32,241,183,142,130,193,32,241    <180>
10020 DATA 183,138,41,1,10,10,141,134,193,
      173,128,193,174,129,193,172,130,193  <112>
10030 DATA 32,186,255,169,0,32,189,255,32,
      192,255,174,128,193,32,201,255,169   <148>
10040 DATA 63,162,1,141,132,193,142,133,19
      3,160,0,185,137,193,240,6,32,210,255 <063>
10050 DATA 200,208,245,32,225,255,240,87,3
      2,105,193,32,201,192,133,2,238,131   <106>
10060 DATA 193,32,89,193,10,5,2,13,134,193
      ,170,189,151,193,32,210,255,238,131  <194>
10070 DATA 193,173,131,193,201,200,208,222
      ,173,133,193,208,5,173,132,193,240   <241>
10080 DATA 17,173,132,193,56,233,1,141,132
      ,193,176,3,206,133,193,76,93,192,160 <227>
10090 DATA 0,185,146,193,240,6,32,210,255,
      200,208,245,173,128,193,32,195,255   <154>
10100 DATA 32,204,255,96,174,128,193,32,20
      1,255,169,13,32,210,255,160,1,76,164 <127>
10110 DATA 192,173,132,193,174,133,193,133
      ,253,134,254,172,131,193,152,41,248  <028>
10120 DATA 141,136,193,133,251,169,0,133,2
      52,6,251,38,252,6,251,38,252,165,251 <032>
10130 DATA 24,109,136,193,133,251,165,252,
      105,0,133,252,6,251,38,252,6,251,38  <166>
10140 DATA 252,6,251,38,252,152,41,7,24,10
      1,251,133,251,165,252,105,0,133,252  <042>
10150 DATA 165,253,24,41,248,101,251,133,2
      51,165,254,101,252,133,252,173,126   <001>
10160 DATA 193,24,101,251,133,251,173,127,
      193,101,252,133,252,165,253,41,7,73  <131>
10170 DATA 7,170,169,1,202,48,3,10,208,250
      ,141,135,193,162,52,120,134,1,160    <094>
10180 DATA 0,177,251,162,55,134,1,88,44,13
      5,193,208,2,160,1,152,96,165,251,24  <096>
10190 DATA 105,1,133,251,165,252,105,0,133
      ,252,76,66,193,169,0,141,131,193,169 <132>
10200 DATA 13,32,210,255,160,18,169,32,32,
      210,255,136,208,250,96,0,32,1,4,1    <048>
10210 DATA 0,0,0,0,0,0,27,64,15,27,51,4,27
      ,56,0,12,27,64,0,0,231,227,225,224   <051>
10220 DATA 224,225,227,231,0               <118>
```


Listing »Superposter 64«. Bitte beachten Sie den Checksummer.



Beispiel einer Hardcopy auf dem Star Gemini

Hardcopy Itoh 8510 mit Hi-Eddi

Besitzern des Itoh 8510 bietet dieses Programm die Möglichkeit, von Hi-Eddi Hardcopies zu erzeugen.

Mit dem Hi-Eddi aus der Ausgabe 1/85 vom 64'er lassen sich mit minimaler Hardware-Ausrüstung (C 64, Diskettenlaufwerk 1541, Joystick) sehr komfortable Grafiken erzeugen. Doch was nützt die schönste Grafik auf Bildschirm oder Diskette, wenn man sie nicht ausdrucken kann?

Die in Ausgabe 1/85 veröffentlichte Druckroutine Hi-Print arbeitet leider nur mit Epson-Druckern zusammen. Da der Itoh 8510 ein verbreiteter und guter Drucker ist, wurden für ihn die im folgenden beschriebenen Programme erstellt. Er wird über eine Centronics-Schnittstelle via User-Port am C 64 angeschlossen.

Zusätzlich wurden einige Features eingebaut, die die Epson-Hardcopy-Routine nicht besitzt. So wäre die Vorgabe von Leerspalten vom linken Rand aus (falls noch Platz ist) eine nützliche Sache. Um zwei Grafiken in der Horizontalen aneinander zu setzen, wurde dazu die volle Bildschirmbreite in 40 Spalten zu je 8 (40 x 8 = 320) Punkten (wie im Textmodus) organisiert. Um jetzt zwei Grafiken nebeneinander zu plazieren, kann dazu die Startspalte (wird mitgedruckt) der links stehenden Grafik (Grafik Nr. 1) und die Endspalte (wird nicht mitgedruckt) der rechts stehenden Grafik (Grafik Nr. 2) vorgewählt werden.

Die Ausdruckbreite wird dazu in drei Sektoren von links nach rechts aufgeteilt.

0. Sektor: Abstand (SPC) in Punkten vom linken Papierrand.
1. Sektor: Grafik Nr. 1 (Bild-Nr. ; Startspalte)
   Bild-Nr. = 0 => kein Bild, Startspalte = 48
   Bild wird rechtsbündig gedruckt
   0 <= Startspalte <= 39
   Startspalte = 0 => volle Hardcopy von Grafik Nr. 1
2. Sektor: Grafik Nr. 2 (Bild-Nr. ; Endspalte)
   Bild-Nr. = 0 => kein Bild, Endspalte = 8
   Bild wird linksbündig gedruckt
   1 <= Endspalte <= 40
   Endspalte = 40 => volle Hardcopy von Grafik Nr. 2

Insgesamt kann folgende Formel für die Aufteilung der drei Sektoren angegeben werden:

SPC + (40—Startspalte) * 8 + Endspalte * 8 = 640

Im Steuerprogramm werden nacheinander die Bild-Nr. von Sektor 1, dann (falls Bild-Nr. <> 0) die Startspalte von Grafik-Nr. 1, dann die Bild-Nr. von Sektor 2 und (falls Bild-Nr. <> 0) die Endspalte von Grafik-Nr. 2, und schließlich die Anzahl der Leerspalten vor Grafik-Nr. 1 abgefragt.

Die Maximalwerte werden stets durch das Steuerprogramm vorgegeben, so daß in vielen Fällen ein einfaches RETURN genügt.

Um die Druckroutine namens »HI-P(ITOH8510)« für den ITOH 8510 in das Steuerprogramm Hi-Eddi einzubilden, mußte dieses auf die Bedürfnisse der Druckroutine angepaßt werden und heißt nun »HI-E(ITOH8510)«.

Dazu wurden (aus Speicherplatzgründen) alle Zeilen neu numeriert und die Textausgabepassagen verkürzt. In den Pro-

grammzeilen 24 und folgende befindet sich das Drucksteuerprogramm.

Es wird hierbei dringend davor gewarnt, das neue Steuerprogramm HI-E(ITOH8510) zu erweitern oder zu ändern, da sonst das Steuerprogramm (inklusive Variablen) nicht mehr in den verfügbaren Basic-Speicherplatz paßt. Dies gilt auch für den eingebauten Parametercheck, da in der Druckroutine selber keine Plausibilitätsprüfung mehr vorgenommen wird.

Für die Initialisierung des Druckers und die Ausgabe der Hardcopy sollte sich der Drucker im SELECT-Modus befinden, da sonst das Programm hängenbleibt (Acknowledge-Signal wird erwartet).

Falls aus irgendeinem Grunde (Drucker beziehungsweise Kabel defekt, falsch oder gar nicht angeschlossen) dieser Fall eintritt, kann mit RUN/STOP-RESTORE die Druckroutine beendet werden. Hi-Eddi kann dann wieder mit RUN gestartet werden. (Das RUN ist notwendig, um HI-EXE nachzuladen, da aus Speicherplatzgründen nach der Anwahl des Drucksteuerprogramms immer die Druckroutine über HI-EXE nachgeladen wird.)

Der Drucker sollte mittels der DIP-Schalter auf CR = CR und 8-Bit-Data eingestellt sein. Da am Anfang kein Carriage Return ausgegeben wird, sollte man dafür sorgen, daß der Druckkopf sich auf einer freien Zeile befindet. Außerdem sollte das Papier zentriert werden, da sich der Ausdruck über die volle Papierbreite erstrecken kann.

Die Anpassung des ITOH 8510A an Hi-Eddi besteht im wesentlichen aus zwei Schritten:

1. HI-P(ITOH8510).L abtippen und starten. Dieses ist ein Ladeprogramm, das die Druckroutine HI-P(ITOH8510) auf Diskette ablegt. Es wird (nach erfolgreichem Test) zum Arbeiten mit Hi-Eddi nicht mehr benötigt.
2. Hi-E(ITOH8510) abtippen und speichern. Es ist das angepaßte Steuerprogramm und ersetzt das alte Hi-Eddi.

So, das wär's eigentlich. Hi-Eddi mit Itoh 8510A ist nun lauffähig. Mir bleibt nur noch, viel Erfolg mit Hi-Eddi zu wünschen.

(Frank Ziesche/rg)

```
10 REM #### PROGRAMM ZUR ERZEUGUNG #####   <190>
20 REM ## VON HI-P(ITOH8510) AUF DISK ##   <124>
30 REM AUTOR: F.ZIESCHE' AM: 13. 2.85 ##   <068>
40 RESTORE:PS=0:REM DATA'S UEBERPRUEFEN    <136>
50 PRINT"{CLR,DOWN,SPACE}DATA-ZEILEN WERDE
   N UEBERPRUFT !"                         <100>
60 READ BY : IF BY < 0 THEN 80             <145>
70 PS = PS + BY : GOTO 60                  <016>
80 IF PS+BY <> 0 THEN PRINT"FEHLER IN DATA
   -ZEILEN !":LIST 160-                    <036>
90 INPUT"{DOWN,SPACE}DISKETTE EINGELEGT{3S
   PACE}JA{4LEFT}";IN$:IF IN$<>"JA"THEN 90 <159>
100 RESTORE : OPEN 3,8,3,"HI-P(ITOH8510),P
    ,W"                                    <061>
110 PRINT#3,CHR$(0);CHR$(13);:REM STARTADR
    ESSE                                   <115>
120 READ BY : IF BY < 0 THEN 140           <233>
130 PRINT#3,CHR$(BY); : GOTO 120           <044>
140 CLOSE 3 : PRINT"{DOWN,SPACE}PROGRAMM W
    URDE ERZEUGT !"                        <107>
150 END                                    <152>
160 REM DATA-ZEILEN FUER HI-P(ITOH8510)    <018>
170 DATA 165,1,72,169,55,133,1,173,87,13,3
    2,186,13,141,170,14,173,107,13,32      <217>
180 DATA 186,13,141,223,14,169,0,141,164,1
    4,141,217,14,141,174,13,32,248,13      <145>
190 DATA 32,212,13,169,27,32,111,14,169,70
    ,32,111,14,169,0,160,0,32,47,14        <162>
200 DATA 169,0,141,148,14,169,0,141,201,14
    ,169,27,32,111,14,169,83,32,111        <055>
210 DATA 14,169,0,160,0,32,47,14,169,0,240
    ,16,32,135,14,32,241,14,238,148        <007>
220 DATA 14,173,148,14,201,40,144,240,169,
    0,240,16,32,188,14,32,241,14,238       <092>
230 DATA 201,14,173,201,14,201,0,144,240,2
    4,173,164,14,105,64,141,164,14,173     <187>
240 DATA 170,14,105,1,141,170,14,24,173,21
    7,14,105,64,141,217,14,173,223,14      <185>
250 DATA 105,1,141,223,14,169,13,32,111,14
    ,169,10,32,111,14,238,174,13,169       <197>
260 DATA 0,201,25,176,3,76,42,13,104,133,1
    ,96,162,7,221,149,31,240,4,202,208     <247>
270 DATA 248,232,138,10,10,10,10,10,96,0,0
    ,0,0,0,0,0,0,24,173,123,13,105,40      <020>
280 DATA 56,237,60,13,133,251,169,0,133,25
    2,160,3,6,251,38,252,136,208,249       <132>
290 DATA 165,251,164,252,141,80,13,140,82,
    13,96,169,255,141,3,221,173,2,221      <055>
300 DATA 9,4,141,2,221,173,0,221,9,4,141,0
    ,221,162,0,189,27,14,32,111,14,232     <033>
310 DATA 224,20,144,245,96,127,27,93,27,78
    ,27,84,49,54,27,102,27,76,48,48        <183>
320 DATA 48,27,48,27,62,133,99,132,98,56,1
    62,144,32,73,188,32,221,189,162        <085>
330 DATA 1,189,0,1,240,6,232,224,6,144,246
    ,202,202,142,81,14,56,169,4,233        <181>
340 DATA 0,240,9,168,169,48,32,111,14,136,
    208,248,172,81,14,240,12,162,1,189     <114>
350 DATA 0,1,32,111,14,232,136,208,246,96,
    141,1,221,173,0,221,41,251,141,0       <177>
360 DATA 221,9,4,141,0,221,173,13,221,41,1
    6,240,249,96,120,165,1,72,41,248       <033>
370 DATA 133,1,169,0,133,252,169,0,133,251
    ,160,3,6,251,38,252,136,208,249        <080>
380 DATA 24,165,251,105,0,133,251,165,252,
    105,0,133,252,177,251,153,204,13       <176>
390 DATA 200,192,8,144,246,104,133,1,88,96
    ,120,165,1,72,41,248,133,1,169,0       <202>
400 DATA 133,252,169,0,133,251,160,3,6,251
    ,38,252,136,208,249,24,165,251,105     <237>
410 DATA 0,133,251,165,252,105,0,133,252,1
    77,251,153,204,13,200,192,8,144        <236>
420 DATA 246,104,133,1,88,96,160,128,140,4
    ,15,169,0,141,8,15,140,10,15,162       <203>
430 DATA 7,189,204,13,41,0,240,7,169,0,9,0
    ,141,8,15,78,10,15,202,16,236,173      <206>
440 DATA 8,15,32,111,14,78,4,15,144,215,96 <148>
450 DATA -57076 :REM  PRUEFSUMME           <184>
```


Listing »HI-P(ITOH8510).L«

```
1 IF A<2 THEN A=A+2:LOAD"HI-EXE",8,1       <058>
2 IF A=3 THEN 6                            <207>
3 IF A=4 THEN 25                           <035>
4 INPUT"B-ART";C:POKE 780,C:POKE 56,13:CLR
  :W$="123456":SYS 3328                    <117>
5 IF PEEK(8073)AND 64 THEN A=3:LOAD"MENUE"
  ,8,1                                     <238>
6 SYS 3337:A=PEEK(8064):ON A+1 GOSUB 7,10,
  12,11,18,24,11,11,19,20,12:GOTO 6        <059>
7 OPEN 3,8,0,"$0":GET#3,C$,C$              <217>
8 GET#3,C$,C$,L$,H$:IF C$=""THEN CLOSE 3:W
  AIT 198,1:RETURN                         <148>
9 PRINT 256*ASC(H$+CHR$(0))+ASC(L$+CHR$(0)
  );:SYS 3334:GOTO 8                       <156>
10 PRINT"{CLR,DOWN}SEQ.:":PRINT"{DOWN,2SPA
   CE}"W$:INPUT"{HOME,3DOWN}";W$:IF LEN(W$
   )<2 THEN 10                             <096>
11 RETURN                                  <069>
12 POKE 8115,8:PRINT"{CLR,DOWN,RVSON}G{RVO
   FF}-BILD":IF PEEK(8073)>127 THEN PRINT"
   {RVSON}F{RVOFF}-BILD"                   <152>
13 PRINT"{RVSON}S{RVOFF}PRITE":PRINT"{RVSO
   N}Z{RVOFF}-SATZ":PRINT"{DOWN}->";       <208>
14 GET C$:IF C$<>"G"AND C$<>"F"AND C$<>"S"
   AND C$<>"Z"THEN 14                      <093>
15 PRINT C$:POKE 780,ASC(C$):H$=W$:INPUT"{
   DOWN}F-NAME";W$:SYS 3331:W$=H$:H$=""    <213>
16 OPEN 1,8,15,H$:INPUT#1,A,H$:PRINT A;H$:
   IF A>0 THEN WAIT 198,1                  <003>
17 CLOSE 1:RETURN                          <144>
18 POKE 631,34:POKE 198,1:H$="":INPUT"{CLR
   ,DOWN}";H$:GOTO 16                      <210>
19 A=8075:GOTO 21                          <059>
20 A=8079                                  <160>
21 D=PEEK(8074):C=PEEK(A+D):PRINT"{CLR,DOW
   N}F"2*D+1:PRINT"S-WEITE:"C,             <165>
22 INPUT C:IF C<1 OR C>160 THEN 22         <125>
23 POKE A+D,C:RETURN                       <180>
24 A=4:LOAD"HI-P(ITOH8510)",8,1            <215>
25 INPUT"{CLR,DOWN}S1:B-NR.(0=KB)";C:POKE
   3415,C AND 7:IF C=0 THEN D=40:GOTO 27   <236>
```

Listing »HI-E(ITOH8510)«. Bitte beachten Sie bei der Eingabe den Checksummer.

```
26 INPUT"SSP.(<=39;0=VOLL){2SPACE}0{3LEFT}
   ";D:IF D<0 OR D>39 THEN 26                <222>
27 POKE 3388,D:C=40+D:IF C>40 THEN C=40      <119>
28 INPUT"{DOWN}S2:B-NR.(0=KB)";D:POKE 3435
   ,D AND 7:IF D=0 THEN 30                   <041>
29 PRINT"ESP.(>0;<=";C;"{LEFT})  ";C;:INPUT
   "{5LEFT}";D:IF D<1 OR D>C THEN 29         <191>
30 POKE 3451,D:C=640-(40-PEEK(3388)+D)*8     <024>
31 PRINT"{DOWN}SPC(=>0;<=";C;"{LEFT}){2SPA
   CE}";C;:INPUT"{6LEFT}";D:IF D<0 OR D>C
   THEN 31                                   <194>
32 C=INT(D/256):POKE 3383,C:INPUT"{DOWN}DR
   UCK{2SPACE}J{3LEFT}";C$:IF C$<>"J"THEN
   34                                        <106>
33 POKE 3381,D-C*256:SYS 3328:INPUT"{DOWN}
   NOCHMAL{2SPACE}J{3LEFT}";C$:IF C$="J"TH
   EN 25                                     <058>
34 A=1:GOTO 1                                <016>
```

© 64'er

Listing »HI-E(ITOH 8510)«. Bitte beachten Sie bei der Eingabe den Checksummer.

Hardcopy von Hi-Eddi auf dem Plotter VC 1520

Mit diesem Programm ist es möglich, Hardcopies von Hi-Eddi mit dem VC 1520 zu erstellen.

Geben Sie zuerst »Copy 1520« (Listing 1) mit dem MSE ein und speichern Sie es ab. Jetzt muß noch das Steuerprogramm geändert werden. Bei dem Steuerprogramm (Listing 2) sind die Zeilen 600 bis 750 zu ändern.

Die Hardcopy kann jetzt von Hi-Eddi per Tastendruck oder Menü nachgeladen werden. Es ist möglich, mehrere Bilder (nahtlos) untereinander oder übereinander (Overlay) zu drucken.

(Stephan Paetzold/rg)


```
50 IF A<2 THEN A=A+2:LOAD"HI-EXE",8,1        <107>
60 IF A=3 THEN 150                           <027>
70 IF A=4 THEN 650                           <200>
100 INPUT"{CLR}BETRIEBSART";C:POKE 780,C:P
    OKE 56,13:CLR:W$="123456":SYS 3328        <254>
110 IF PEEK(8073)AND 64 THEN A=3:LOAD"MENU
    E",8,1                                    <087>
150 SYS 3337:A=PEEK(8064)                     <119>
160 ON A+1 GOSUB 200,250,300,290,400,600,2
    90,290,500,510,300:GOTO 150               <254>
200 OPEN 3,8,0,"$0":GET#3,C$,C$               <154>
210 GET#3,C$,C$,L$,H$:IF C$=""THEN CLOSE 3
    :WAIT 198,1:RETURN                        <094>
220 PRINT 256*ASC(H$+CHR$(0))+ASC(L$+CHR$(
    0));:SYS 3334:GOTO 210                    <018>
250 PRINT"{CLR,DOWN}SEQUENZ:":PRINT"{DOWN,
    2SPACE}"W$:INPUT"{HOME,3DOWN}";W$:IF L
    EN(W$)<2 THEN 250                         <062>
290 RETURN                                    <094>
300 POKE 8115,8:PRINT"{CLR,DOWN}G=GRAFIKBI
    LD":IF PEEK(8073)>127 THEN PRINT"F=FAR
    BBILD"                                    <036>
310 PRINT"S=SPRITE":PRINT"Z=ZEICHENSATZ":P
    RINT"{DOWN,SPACE}-->";                    <017>
320 GET C$:IF C$<>"G"AND C$<>"F"AND C$<>"S
    "AND C$<>"Z"THEN 320                      <240>
330 PRINT C$:POKE 780,ASC(C$):H$=W$:INPUT"
    {DOWN}FILENAME";W$:SYS 3331:W$=H$:H$="
    "                                         <046>
340 OPEN 1,8,15,H$:INPUT#1,A,H$:PRINT A;H$
    :IF A>0 THEN WAIT 198,1                   <073>
350 CLOSE 1:RETURN                            <223>
400 POKE 631,34:POKE 198,1:H$="":INPUT"{CL
    R,DOWN}";H$:GOTO 340                      <207>
500 A=8075:GOTO 520                           <053>
510 A=8079                                    <140>
520 D=PEEK(8074):C=PEEK(A+D):PRINT"{CLR,DO
    WN}F"2*D+1:PRINT"SCHRITTWEITE :"C,        <055>
530 INPUT C:IF C<1 OR C>160 THEN 530          <066>
540 POKE A+D,C:RETURN                         <189>
600 IF A=5 THEN A=4:LOAD"COPY 1520",8,1       <045>
650 INPUT"{CLR}BILDNUMMER";D:C=(D AND 7)      <212>
660 INPUT"FARBE (0-3)";B$:POKE 1000,0:POKE
    1001,VAL(B$)                              <099>
670 FOR I=0 TO 6                              <103>
680 IF PEEK(8086+I)=D THEN LET Z=(I+1)*819
    2                                         <162>
690 NEXT I                                    <010>
700 X=INT(Z/256):POKE 3399,X:POKE 3392,Z-X
    *256                                      <025>
710 POKE 780,C:SYS 3328                       <028>
720 PRINT"OVERLAY";:GOSUB 800:IF C$="J"THE
    N 630                                     <159>
730 PRINT"NAHTLOS";:GOSUB 800:IF C$="N"THE
    N LET A=1:GOTO 50                         <161>
740 OPEN 1,6,1:PRINT#1,"M",0,-180:CLOSE 1     <106>
750 OPEN 1,6,0:PRINT#1," ":CLOSE 1:GOTO 65
    0                                         <077>
800 PRINT" ? (J/N)";                          <205>
810 GET C$:IF C$<>"J"AND C$<>"N"THEN 810      <210>
820 PRINT C$:RETURN                           <230>
```

© 64'er

Listing 2. »Steuerprogramm Hi-Eddi«. Bitte beachten Sie bei der Eingabe den Checksummer.

```
 programm : copy 1520          0d00 103b
-----------------------------------------

0d00 : a2 00 8e 0e dc 78 86 b7   23
0d08 : e8 86 b8 86 b9 a2 06 86   08
0d10 : ba 20 c0 ff a2 02 86 b8   d0
0d18 : 86 b9 a2 00 86 b7 a2 06   e0
0d20 : 86 ba 20 c0 ff a2 00 8e   56
0d28 : 40 03 8e 41 03 8e 43 03   6d
0d30 : 8e 46 03 8e 47 03 8e 48   cb
0d38 : 03 8e 49 03 8e 4c 03 a2   d2
0d40 : 00 86 f7 8e 94 0e a2 e0   59
0d48 : 86 f8 8e 98 0e a2 01 8e   18
0d50 : 4b 03 a2 d0 8e a3 0d 8e   37
0d58 : d7 0d a2 02 20 c9 ff ad   4a
0d60 : e9 03 18 69 30 20 d2 ff   4d
0d68 : a9 0d 20 d2 ff 20 cc ff   2e
0d70 : a9 34 85 01 a0 00 b1 f7   76
0d78 : a2 37 86 01 48 18 a5 f7   43
0d80 : 69 08 90 02 e6 f8 85 f7   8e
0d88 : ad e8 03 f0 38 68 a2 00   da
0d90 : 8e 4a 03 0a 2e 4a 03 0a   9b
0d98 : 2e 4a 03 48 aa ae 4a 03   04
0da0 : ec 4b 03 d0 03 20 00 0f   5c
0da8 : ee 49 03 ee 41 03 d0 03   4f
0db0 : ee 40 03 ee 41 03 d0 03   d2
0db8 : ee 40 03 ad 49 03 c9 04   19
0dc0 : d0 cb 4c ee 0d 68 a2 00   05
0dc8 : 8e 4a 03 18 0a 2e 4a 03   80
0dd0 : 48 ad 4a 03 cd 4b 03 d0   c7
0dd8 : 03 20 00 0f ee 49 03 ee   f0
0de0 : 41 03 d0 03 ee 40 03 ad   90
0de8 : 49 03 c9 08 d0 d7 a2 00   7d
0df0 : 8e 49 03 68 ee 48 03 ad   89
0df8 : 48 03 c9 28 b0 03 4c 70   6e
0e00 : 0d ad 4c 03 f0 03 20 00   ff
0e08 : 0f ee 43 03 ad 4e 03 8d   34
0e10 : 41 03 ad 4d 03 8d 40 03   8b
0e18 : 20 00 0f a2 00 8e 48 03   ec
0e20 : 8e 40 03 8e 41 03 8e 4c   60
0e28 : 03 a9 d0 8d a3 0d 8d d7   6e
0e30 : 0d c6 f8 a5 f7 38 e9 3f   fa
0e38 : b0 02 c6 f8 85 f7 ee 47   1c
0e40 : 03 ad 47 03 c9 08 f0 03   f3
0e48 : 4c 70 0d a2 00 8e 47 03   fc
0e50 : e6 f8 a5 f7 18 69 38 90   e9
0e58 : 02 e6 f8 85 f7 ee 46 03   d2
0e60 : ad 46 03 c9 19 f0 03 4c   e8
0e68 : 70 0d 20 cc ff ad e8 03   17
0e70 : d0 03 4c f6 0e ee 4b 03   3f
0e78 : ad 4b 03 c9 04 f0 77 a2   b0
0e80 : 00 8e 46 03 8e 43 03 8e   e6
0e88 : 40 03 8e 41 03 8e 4d 03   f5
0e90 : 8e 4e 03 a2 00 86 f7 a2   b4
0e98 : e0 86 f8 a2 01 20 c9 ff   86
0ea0 : a2 00 8a 48 bd e7 0e 20   82
0ea8 : d2 ff 68 aa e8 e0 09 d0   45
0eb0 : f1 a2 01 20 c9 ff a2 00   5e
0eb8 : 8a 48 bd f0 0e 20 d2 ff   21
0ec0 : 68 aa e8 e0 06 d0 f1 a2   c7
0ec8 : 02 20 c9 ff ad 4b 03 38   fe
0ed0 : e9 01 a8 b9 e9 03 18 69   85
0ed8 : 30 20 d2 ff a9 0d 20 d2   f6
```

Listing 1. »Copy 1520«. Bitte beachten Sie bei der Eingabe den MSE.

```
0ee0 : ff 20 cc ff 4c 70 0d 4d  39
0ee8 : 2c 30 2c 2d 32 30 31 0d  61
0ef0 : 4d 2c 30 2c 30 0d 20 2f  2f
0ef8 : f3 a9 01 8d 0e dc 58 60  9c
0f00 : ad 40 03 48 ad 41 03 48  19
0f08 : ad 43 03 48 a2 00 8e 43  0c
0f10 : 03 8e 44 03 8e 45 03 c9  7f
0f18 : 64 90 08 e9 64 ee 43 03  d4
0f20 : 4c 17 0f c9 0a 90 08 e9  0e
0f28 : 0a ee 44 03 4c 23 0f c9  c9
0f30 : 01 90 08 e9 01 ee 45 03  5b
0f38 : 4c 2f 0f ad 41 03 ae 40  fd
0f40 : 03 f0 11 a2 02 8e 40 03  f0
0f48 : a2 05 8e 41 03 e8 8e 42  6f
0f50 : 03 4c 5f 0f a2 00 8e 40  18
0f58 : 03 8e 41 03 8e 42 03 c9  ee
0f60 : 64 90 08 ee 40 03 e9 64  79
0f68 : 4c 5f 0f c9 0a 90 08 ee  84
0f70 : 41 03 e9 0a 4c 6b 0f c9  de
0f78 : 01 90 08 ee 42 03 e9 01  87
0f80 : 4c 77 0f ad 41 03 18 69  61
0f88 : 08 8d 41 03 ad 42 03 c9  94
0f90 : 0a 90 08 e9 0a 8d 42 03  3e
0f98 : ee 41 03 ad 41 03 c9 0a  05
0fa0 : 90 0b e9 0a 8d 41 03 ee  3e
0fa8 : 40 03 4c 9b 0f a2 01 20  3b
0fb0 : c9 ff ad 4c 03 f0 08 a9  99
0fb8 : 44 20 d2 ff 4c c4 0f a9  3b
0fc0 : 4d 20 d2 ff a9 2c 20 d2  f4
0fc8 : ff ad 40 03 18 69 30 20  dc
0fd0 : d2 ff ad 41 03 18 69 30  2c
0fd8 : 20 d2 ff ad 42 03 18 69  86
0fe0 : 30 20 d2 ff a9 2c 20 d2  f7
0fe8 : ff a9 2d 20 d2 ff ad 43  76
0ff0 : 03 18 69 30 20 d2 ff ad  54
0ff8 : 44 03 18 69 30 20 d2 ff  40
1000 : ad 45 03 18 69 30 20 d2  52
1008 : ff a9 0d 20 d2 ff 68 8d  0d
1010 : 43 03 68 8d 41 03 68 8d  8a
1018 : 40 03 ad a3 0d 49 20 8d  70
1020 : a3 0d 8d d7 0d ad 4c 03  1e
1028 : 49 01 8d 4c 03 ad 41 03  87
1030 : 8d 4e 03 ad 40 03 8d 4d  48
1038 : 03 60 34 ff ff ff ff ff  78
```

Listing 1. »Copy 1520« (Schluß). Bitte beachten Sie bei der Eingabe den MSE.

Hardcopy MPS 802/1526

Hardcopies in der vierfachen Größe erstellt dieses Programm auf dem MPS 802.

Dieses Programm druckt den HiRes-Grafikbildschirm in vierfacher Vergrößerung auf dem MPS 802 aus. Durch die Papierbreite bedingt, muß das Bild jedoch in zwei Teile zerlegt und anschließend zusammengeklebt werden, was sich jedoch nicht nachteilig bemerkbar macht.

Die Schnelligkeit der Druckroutine (komplett zirka 8 bis 13 Minuten) wird durch zwei Faktoren bewirkt:

1. Adreßberechnung und Datenumwandlung in Maschinensprache;

2. Keine Verwendung von selbstdefinierten Sonderzeichen, das heißt die gedruckten Zeichen sind die Standard-Grafikzeichen des Commodore 64 (es handelt sich hierbei um folgende 16 Zeichen:
CHR$(160),(161),(162),(172),(187),(188),(190),(191)
(in normaler und reverser Darstellung)

Hardcopy des Hi-Eddi-Menüs

Während des Druckes ist auf den HiRes-Bildschirm umgeschaltet, so daß man die Abarbeitung verfolgen kann. Nach Fertigstellung des ersten Teiles unterbricht das Programm, um ein Neujustieren des Druckerpapiers zu ermöglichen. Danach wird der zweite Teil ausgedruckt. Die Bilder sollten sich im Speicher ab Adresse 8192 = $2000 befinden.

(Ralph Hoffmann/Christoph Bertog/rg)

```
1 REM **********************************    <132>
2 REM   4-FACH VERGROESSERTE HARDCOPY        <010>
3 REM              BY                        <232>
4 REM          RALPH HOFFMANN                <144>
5 REM         CHRISTOPH BERTOG               <002>
6 REM         LIEBKNECHTSTR.17               <151>
7 REM           ZWEIBRUECKEN                 <193>
8 REM                                        <070>
9 REM **********************************    <140>
10 FOR Q=52224 TO 52540:READ D:POKE Q,D:X=
   X+D:NEXT                                  <069>
15 IF X<>42320 THEN PRINT"DATA-FEHLER":END <008>
20 PRINT"{CLR}":A1=PEEK(53265):A2=PEEK(532
   72)                                       <146>
25 OPEN 6,4,6:PRINT#6,CHR$(20)               <168>
30 OPEN 4,4:PRINT#4,CHR$(141)                <196>
35 POKE 254,0:POKE 255,32                    <165>
50 POKE 53265,59:POKE 53272,24               <071>
55 SYS 52224                                 <105>
60 POKE 53265,A1:POKE 53272,A2               <162>
65 IF P=1 THEN 100                           <083>
70 INPUT"NEU JUSTIERT";C$                    <030>
75 IF C$<>"J"THEN 97                         <048>
[illegible] POKE 254,160:POKE 255,32:P=1:PRINT"{CLR
   }":GOTO 50                                <185>
100 CLOSE 6:CLOSE 4:END                      <248>
32000 DATA 169,160,141,224,3,141,239,3,169
      ,172,141,225,3,141,238,3,169,187,141 <018>
32001 DATA 226,3,141,237,3,169,162,141,227
      ,3,141,236,3,169,188,141,228,3,141,2
      35                                     <043>
32002 DATA 3,169,161,141,229,3,141,234,3,1
      69,191,141,230,3,141,233,3,169,190,1
      41                                     <235>
32003 DATA 231,3,141,232,3,169,0,141,240,3
      ,141,241,3,169,1,141,244,3,141,246,3 <064>
32004 DATA 169,0,141,245,3,162,4,142,242,3
      ,32,201,255,169,0,141,243,3,173,240  <248>
32005 DATA 3,24,109,241,3,168,177,254,10,4
      6,243,3,10,46,243,3,145,254,200,177  <167>
32006 DATA 254,10,46,243,3,10,46,243,3,145
      ,254,206,242,3,240,3,76,193,204,169  <177>
32007 DATA 4,141,242,3,169,8,24,109,240,3,
      141,240,3,201,160,208,235,169,0,141  <097>
32008 DATA 240,3,169,2,24,109,241,3,141,24
      1,3,201,8,208,217,169,0,141,241,3,16
      9                                      <113>
32009 DATA 64,24,101,254,133,254,144,2,230
      ,255,230,255,173,243,3,201,11,176,8  <013>
32010 DATA 201,5,144,33,201,8,176,29,169,1
      8,32,210,255,172,243,3,185,224,3,32  <022>
32011 DATA 210,255,173,246,3,201,20,240,17
      ,169,146,32,210,255,76,246,204,172,2
      43                                     <034>
32012 DATA 3,185,224,3,32,210,255,238,246,
      3,173,246,3,201,21,240,3,76,93,204,1
      69                                     <191>
32013 DATA 1,141,246,3,169,20,24,109,245,3
      ,141,245,3,201,80,240,16,170,169,141 <147>
32014 DATA 32,210,255,169,32,32,210,255,20
      2,208,250,240,219,169,13,32,210,255  <234>
32015 DATA 169,0,141,245,3,238,244,3,173,2
      44,3,201,101,208,199,32,204,255,96   <198>
```


Listing »Hardcopy MPS 802«. Bitte beachten Sie bei der Eingabe den Checksummer.

Hardcopy in doppelter Größe

Mit diesem Programm ist es jetzt auch MPS 801-Besitzern möglich, Hardcopies in doppelter Größe auszugeben.

»BIGCOPY 64« wurde abgefaßt, um auch mit dem Commodore MPS 801 eine Hardcopy in doppelter Größe anfertigen zu können.

Was uns besonders störte, war die Tatsache, daß es anscheinend keine Programme gab, die auf Siebennadeldruckern Hardcopies in doppelter Größe ermöglichten. Da wir jedoch unsere Grafiken zur Abwechslung einmal in akzeptabler Größe haben wollten, sahen wir uns gezwungen, ein eigenes Programm zu schreiben. Im folgenden wollen wir von den Schwierigkeiten, die sich uns in den Weg stellten und deren Beseitigung berichten.

Betrachtet man einmal die Speichernutzung des C 64 bei Grafiken, so ist ersichtlich, daß sich die Einteilung in horizontal liegende Bytes ideal für Hardcopies eignet, die auf Achtnadeldruckern ausgegeben werden sollen. Der Commodore MPS 801-Matrixdrucker hat aber die unangenehme Eigenschaft, die Grafiken mit lediglich sieben Nadeln auf das Papier zu bringen. Da man mit dem MPS 801 lediglich 480 Punkte in einer Zeile drucken kann, ist man bei der Erstellung von Programmen, die doppelt große Hardcopies ermöglichen sollen, darauf angewiesen, die Grafiken nicht wie üblich quer auszugeben, sondern sie müssen längs ausgedruckt werden. In dieser Richtung kann man die 400 x 640 Punkte problemlos auf das Papier bringen.

Jetzt aber stellt sich das oben angesprochene Problem: Wie die acht Bit eines Bytes mit sieben Nadeln ausdrucken? Die Antwort ist leider klar: Gar nicht, es geht nämlich nicht! Also muß ein Algorithmus her, der es gestattet, das bei ersten Versuchen unterschlagene achte Bit dennoch auszudrucken. Dieser Algorithmus muß aber, um eben dieses achte Bit zu berücksichtigen, Punkt für Punkt des zu druckenden Bildes abfragen, um so die gesamte Grafik in vertikale Siebenerscheibchen zu zerlegen, die der MPS 801 anstandslos verarbeitet. Durch diese, zugegebenermaßen recht unelegante, aber nötige Prozedur, geht jedoch der oben erwähnte positive Effekt verloren, den die interne Grafikspeicherung auf Berechnungen haben kann. Aus diesem Grund dauert eine Hardcopy mit der Basic-Version von »BIGCOPY« auch zirka 150 Minuten, eine nicht gerade berauschende Zeit für den Erhalt einer einzigen Grafik. Was liegt bei solcherart trödelnden Programmen näher, als sie in der um ein vielfaches schnelleren Maschinensprache zu verfassen.

Hier sei eine kurze Bemerkung über die Verträglichkeit des Programmes mit anderen Druckern als dem MPS 801 gesagt. Die Basic-Version läßt sich relativ leicht auf andere Drucker umschreiben, lediglich die OPEN-Anweisungen und die CHR$-Codes müssen geändert werden. Dies gilt selbstverständlich nur für die unglücklichen Besitzer von Siebennadeldruckern, die bisher gezwungen waren, ihre Grafiken in »Briefmarkenformat« ausdrucken zu lassen.

Die Maschinensprachversion des Programmes gibt sich auch schon viel freundlicher, was den Zeitbedarf angeht, sie liefert eine Hardcopy in einem Elftel der Zeit des Basic-

Hardcopy in 13 Minuten

Programmes, das heißt daß man jetzt nur noch 13 Minuten auf seine Grafik zu warten braucht. Wundert man sich jetzt, warum die als bis zu hundert Mal schnellere Maschinensprache lediglich eine Reduktion des Zeitbedarfs um 91% bringt, so sei man daran erinnert, daß der Drucker ja auch nicht gerade zu den schnellsten gehört, er ist also das bremsende Glied.

Da wir unser Programm als vollständiges Maschinenprogramm verfaßten, mußten aus Komfortgründen gewisse Änderungen vorgenommen werden. Daraus resultierte die vorliegende Form in zwei einzutippenden Teilprogrammen, die ein drittes, funktionsfähiges »BIGCOPY« erzeugen.

Es folgt die Anleitung, wie die Programme eingegeben werden müssen. Zuerst werden beide Listing »BIGCOPY-DATAs« (Listing 1) und »BIGCOPY-Erzeuger« (Listing 2) eingetippt und ohne sie zu starten auf Diskette abgespeichert. Nach der Speicherung des DATA-Teiles sollte dieser Teil jedoch probehalber gestartet werden, um eventuelle Tippfehler durch die Prüfsummenbildung zu identifizieren und zu korrigieren. Diese verbesserte Version muß dann erneut abgespeichert werden. Bei dem Erzeugerprogramm dürfen die REM-Zeilen auf keinen Fall mit eingetippt werden! Zum Erzeugen von »BIGCOPY 64« lädt man zuerst das DATA-Programm, startet es und lädt danach den Erzeugerteil. Nachdem eine Diskette eingelegt wurde, auf der mindestens neun Blöcke Platz sein müssen, wird der Erzeuger mit RUN 1000 gestartet. Das erzeugte »BIGCOPY« kann dann einfach mit LOAD "BIGCOPY 64",8 geladen und mit RUN gestartet werden. Diese etwas umständliche Prozedur mußte gewählt werden, da der Maschinenteil des Programmes direkt hinter dem Basic-Teil liegen muß, um alle sieben Grafikspeicher ausdrucken zu können.

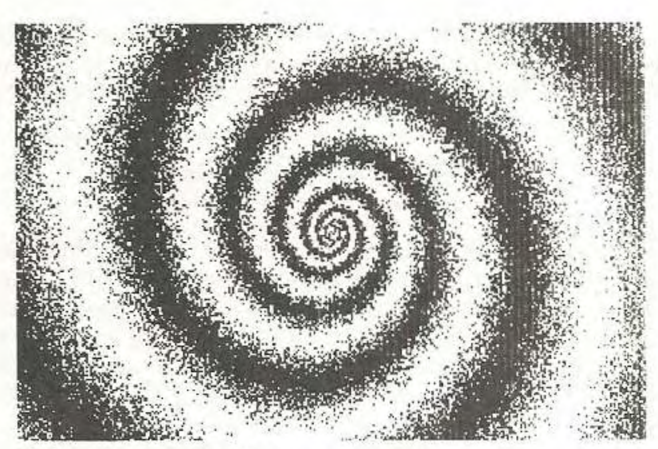
Speicheraufteilung

Jetzt zum Programm an sich.

Um alle sieben Grafikspeicher ansprechen zu können, muß auf jeden Fall mit Maschinensprache gearbeitet werden, da ein Ausschalten des Basic-ROMs von Basic aus logischerweise Selbstmord wäre.

Im Maschinenteil ist deswegen eine Vertauschroutine, die das auszudruckende Bild in den Bereich $2000-$3FFF verschiebt. Diese Routine wird mit Basic angesprochen und wechselt den gewählten Grafikspeicher mit dem ab $2000 aus. Nachdem die Grafik eingeschaltet ist, wird die Druckroutine aufgerufen, die im Maschinensprachteil hinter der Tauschroutine liegt. Nach erfolgtem Ausdruck werden die beiden Bereiche (gewählter und $2000-$3FFF) wieder zurückgeschrieben, so daß kein Bild verlorengeht. Auf diese Weise können alle sieben Grafikspeicher hintereinander ausgedruckt werden. Die Maschinenroutine für den Ausdruck (Listing 3) wurde fast ohne Änderungen aus der Basic-Version der Druckroutine gebildet. Die Basic-Routine verdeutlicht den Algorithmus und vereinfacht die Anpassung an andere Siebennadeldrucker. Zusätzlich läßt sie sich als Unterprogramm in eigenen Programmen gebrauchen.

Da unser Maschinenprogramm bei $0D00 beginnt, und das im Januarheft des 64'er veröffentlichte Hi-Eddi zufälligerweise auch eine Druckroutine benötigt, die ab $0D00 liegt, kamen wir auf die Idee, »BIGCOPY« an Hi-Eddi anzupassen.



Beispiel einer Hardcopy.
Das Original ist 25,7x17 cm groß.

Für die Anpassung an Hi-Eddi sind zwei Teilschritte notwendig. Zum ersten müssen im Hi-Eddi-Steuerprogramm (Listing 3 im 64'er, Ausgabe 1/85) die Zeilen 600-820 (Listing 4) durch die Änderungen ersetzt werden, zweitens muß ein Hi-Print-Programm (Listing 5) erzeugt werden, das aus unserer Maschinenroutine besteht. Zur Erzeugung von Hi-Print werden die Basic-Zeilen an den DATA-Teil des »BIGCOPY«-Listings angehängt, danach wird der DATA-Teil einfach mit RUN gestartet, dadurch wird Hi-Print auf Diskette geschrieben.

(M. Kowall/E. Wilde/rg)

```
5 REM -----BLOCK 1-----(7213)                      <051>
10 DATA 120,165,1,41,0,133,1,169,32,133,6,
   169,224,133,4,169,0,133,5,133,3                 <152>
20 DATA 162,32,160,0,177,3,133,9,177,5,145
   ,3,165,9,145,5,200,208,241,230                  <005>
30 DATA 4,230,6,202,208,234,165,1,9,55,133
   ,1,88,96,169,1,141,11,17,169,63                 <245>
40 DATA 141,10,17,169,0,141,12,17,174,12,1
   7,32,59,14,173,16,17,157,32,17                  <153>
45 REM -----BLOCK 2-----(8794)                     <174>
50 DATA 173,17,17,157,32,19,232,173,16,17,
   157,32,17,173,17,17,157,32,19,238               <112>
60 DATA 12,17,232,173,12,17,201,100,208,21
   7,162,0,32,59,14,173,16,17,157                  <076>
70 DATA 32,18,173,17,17,157,32,20,232,173,
   16,17,157,32,18,173,17,17,157,32                <086>
80 DATA 20,238,12,17,232,173,12,17,201,200
   ,208,217,32,46,16,32,96,16,169                  <131>
90 DATA 0,141,18,17,174,18,17,189,32,17,32
   ,210,255,238,18,17,173,18,17,201                <167>
95 REM -----BLOCK 3-----(10886)                    <209>
100 DATA 200,208,237,169,0,141,18,17,174,1
    8,17,189,32,18,32,210,255,238,18               <039>
110 DATA 17,173,18,17,201,200,208,237,169,
    13,32,210,255,169,9,32,210,255,32              <038>
120 DATA 96,16,169,0,141,18,17,174,18,17,1
    89,32,19,32,210,255,238,18,17,173              <202>
130 DATA 18,17,201,200,208,237,169,0,141,1
    8,17,174,18,17,189,32,20,32,210                <019>
140 DATA 255,238,18,17,173,18,17,201,200,2
    08,237,169,13,32,210,255,169,9,32              <209>
145 REM -----BLOCK 4-----(8796)                    <083>
150 DATA 210,255,173,10,17,56,233,7,141,10
    ,17,201,249,240,7,201,253,240,11               <218>
160 DATA 76,65,13,169,0,141,11,17,76,65,13
    ,32,204,255,169,4,32,195,255,96                <179>
170 DATA 169,128,141,16,17,141,17,17,173,1
    1,17,141,14,17,141,2,17,173,10,17              <227>
180 DATA 141,13,17,141,1,17,173,12,17,141,
    3,17,32,110,15,173,9,17,240,9,24               <148>
190 DATA 173,16,17,105,3,141,16,17,173,13,
    17,208,12,169,0,141,14,17,169,255              <141>
195 REM -----BLOCK 5-----(6787)                    <197>
200 DATA 141,13,17,208,3,206,13,17,173,13,
    17,141,1,17,173,14,17,141,2,17,173             <016>
210 DATA 12,17,141,3,17,32,110,15,173,9,17
    ,240,9,24,173,16,17,105,12,141,16              <008>
220 DATA 17,206,13,17,173,13,17,141,1,17,1
    73,14,17,141,2,17,173,12,17,141                <053>
230 DATA 3,17,32,110,15,173,9,17,240,9,24,
    173,16,17,105,48,141,16,17,206,13              <063>
240 DATA 17,173,13,17,141,1,17,173,14,17,1
    41,2,17,173,12,17,141,3,17,32,110              <200>
245 REM -----BLOCK 6-----(7229)                    <006>
250 DATA 15,173,9,17,240,12,24,173,16,17,1
    05,64,141,16,17,238,17,17,206,13               <084>
260 DATA 17,208,7,169,128,141,17,17,208,11
    1,173,13,17,141,1,17,173,14,17,141             <035>
270 DATA 2,17,173,12,17,141,3,17,32,110,15
    ,173,9,17,240,9,24,173,17,17,105               <161>
280 DATA 6,141,17,17,206,13,17,173,13,17,1
    41,1,17,173,14,17,141,2,17,173,12              <144>
290 DATA 17,141,3,17,32,110,15,173,9,17,24
    0,9,24,173,17,17,105,24,141,17,17              <020>
295 REM -----BLOCK 7-----(7422)                    <023>
300 DATA 206,13,17,173,13,17,141,1,17,173,
    14,17,141,2,17,173,12,17,141,3,17              <220>
310 DATA 32,110,15,173,9,17,240,9,24,173,1
    7,17,105,96,141,17,17,96,173,1,17              <122>
320 DATA 141,0,17,173,3,17,141,4,'7,173,1,
    17,41,248,141,1,17,173,2,17,'  ,1              <060>
330 DATA 141,2,17,173,4,17,41,248,141,4,17
    ,173,3,17,41,7,141,3,17,169,0,141              <070>
340 DATA 5,17,173,4,17,141,6,17,160,4,24,1
    09,6,17,144,3,238,5,17,136,208,244             <180>
345 REM -----BLOCK 8-----(7523)                    <010>
350 DATA 141,4,17,160,3,24,46,4,17,46,5,17
    ,136,208,246,173,3,17,24,109,4,17              <206>
360 DATA 144,3,238,5,17,24,109,1,17,144,3,
    238,5,17,24,133,2,173,5,17,109,2               <112>
370 DATA 17,105,32,133,3,173,0,17,41,7,141
    ,0,17,169,7,56,237,0,17,141,0,17               <121>
380 DATA 208,10,169,1,141,7,17,173,0,17,24
    0,14,169,1,141,7,17,24,14,7,17,206             <221>
390 DATA 0,17,208,248,24,160,0,177,2,141,8
    ,17,13,7,17,56,237,8,17,208,8,169              <030>
395 REM -----BLOCK 9-----(10566)                   <254>
400 DATA 1,141,9,17,76,45,16,169,0,141,9,1
    7,96,169,4,162,4,160,0,32,186,255              <025>
410 DATA 169,0,32,189,255,32,192,255,169,6
    ,162,4,160,6,32,186,255,162,6,32               <034>
420 DATA 201,255,169,16,32,210,255,32,204,
    255,169,6,32,195,255,162,4,32,201              <054>
430 DATA 255,96,169,8,32,210,255,169,27,32
    ,210,255,169,16,32,210,255,169,0               <234>
440 DATA 32,210,255,169,40,32,210,255,96           <151>
450 S=0:FOR T=1 TO 82:READ A:S=S+A:POKE 49
    152+T,A:NEXT:IF S<>7213 THEN F=1:GOTO
    1000                                           <128>
460 S=0:FOR T=83 TO 180:READ A:S=S+A:POKE
    49152+T,A:NEXT:IF S<>8794 THEN F=2:GOT
    O 1000                                         <183>
470 S=0:FOR T=181 TO 276:READ A:S=S+A:POKE
     49152+T,A:NEXT:IF S<>10886 THEN F=3:G
    OTO 1000                                       <021>
480 S=0:FOR T=277 TO 376:READ A:S=S+A:POKE
     49152+T,A:NEXT:IF S<>8796 THEN F=4:GO
    TO 1000                                        <108>
490 S=0:FOR T=377 TO 480:READ A:S=S+A:POKE
     49152+T,A:NEXT:IF S<>6787 THEN F=5:GO
    TO 1000                                        <234>
500 S=0:FOR T=481 TO 583:READ A:S=S+A:POKE
     49152+T,A:NEXT:IF S<>7229 THEN F=6:GO
    TO 1000                                        <231>
510 S=0:FOR T=584 TO 691:READ A:S=S+A:POKE
     49152+T,A:NEXT:IF S<>7422 THEN F=7:GO
    TO 1000                                        <204>
520 S=0:FOR T=692 TO 801:READ A:S=S+A:POKE
     49152+T,A:NEXT:IF S<>7523 THEN F=8:GO
    TO 1000                                        <033>
530 S=0:FOR T=802 TO 890:READ A:S=S+A:POKE
     49152+T,A:NEXT:IF S<>10566 THEN F=9:G
    OTO 1000                                       <214>
540 PRINT"ALLES OKAY !!!":END                      <058>
1000 PRINT"FEHLER IM BLOCK";F                      <239>
```


Listing 1. »BIGCOPY DATAs«. Bitte beachten Sie bei der Eingabe den Checksummer 64 in diesem Heft.

```
5 REM   WICHTIG !!!                               <171>
6 REM   BEIM EINGEBEN VON BIGCOPY ERZEUGER
   UNBEDINGT DIE REM-ZEILEN WEGLASSEN.            <186>
10 REM  BIGCOPY ERZEUGER ERZEUGT DAS DRUCK
   PROGRAMM BIGCOPY 64 AUF DISKETTE               <206>
20 REM  DAZU LAEDT MAN ZUERST BIGCOPY DATA
   S UND STARTET ES.                              <188>
30 REM  WENN ALLES OKAY IST,LAEDT MAN BIGC
   OPY ERZEUGER EIN,LEGT EINE FREIE               <123>
40 REM  DISKETTE EIN UND STARTE DAS PROGRA
   MM MIT RUN1000.                                <100>
50 REM  NACH CA. 1 MINUTE STEHT BIGCOPY 64
   AUF DISKETTE UND KANN JEDERZEIT                <242>
60 REM  EINGELESEN UND GESTARTET WERDEN.          <024>
100 POKE 55,255:POKE 56,12:REM MASCHINENPR
    OGRAMM SCHUETZEN                              <089>
110 PRINT"{CLR}":PRINT"{7SPACE}-----BIGCOP
    Y 64-----"                                    <080>
120 PRINT"--FUER COMMODORE MPS 801 MATRIXD
    RUCKER--"                                     <134>
130 PRINT:PRINT"DRUCKER ANGESCHLOSSEN UND
    EINGESCHALTET ? (Y/N)"                        <004>
140 GET A$:IF A$<>"Y"THEN 140                     <119>
150 PRINT:PRINT:PRINT"WELCHER DER 7 HIRES
    GRAPHIKSPEICHER SOLL"                         <086>
160 PRINT"MIT BIGCOPY AUSGEDRUCKT WERDEN ?
    (1-7) "                                       <194>
170 GET B$:A=VAL(B$):IF(A<1)OR(A>7)THEN 17
    0                                             <152>
180 PRINT:PRINT"ES WIRD BILD";A;"AUSGEDRUC
    KT!"                                          <232>
190 PRINT:PRINT"JETZT NOCH CA. 13 MINUTEN
    GEDULD!":FOR T=1 TO 1000:NEXT                 <178>
200 A=A*32:POKE 3340,A:POKE 3336,32:REM A=
    BILDADRESSE                                   <059>
210 SYS 3328:REM BILD WIRD NACH $2000 GEBR
    ACHT                                          <157>
220 POKE 53265,59:POKE 53272,24:PRINT"{CLR
    }":REM BILD WIRD ANGEZEIGT                    <152>
230 SYS 3383:REM DRUCKROUTINE BEGINNT             <226>
240 POKE 3340,32:POKE 3336,A                      <043>
250 POKE 53265,155:POKE 53272,21                  <057>
260 SYS 3328:REM BILD WIRD NACH A ZURUECKG
    EBRACHT                                       <140>
270 PRINT"NOCH EIN BILD ? (Y/N)"                  <021>
280 GET Z$                                        <180>
290 IF Z$="Y"THEN GOTO 110                        <041>
300 IF Z$<>"N"THEN 280                            <173>
310 IF Z$="N"THEN SYS 64738                       <044>
400 REM ZEILEN 1000-1040 ERZEUGEN BIGCOPY
    AUF DISKETTE                                  <012>
1000 OPEN 1,8,1,"BIGCOPY 64"                      <161>
1010 PRINT#1,CHR$(1);CHR$(8);                     <180>
1020 FOR T=2049 TO 2687:PRINT#1,CHR$(PEEK(
     T));:NEXT                                    <221>
1030 FOR T=2688 TO 3326:PRINT#1,CHR$(0);:N
     EXT                                          <247>
1040 FOR T=49152 TO 50042:PRINT#1,CHR$(PEE
     K(T));:NEXT                                  <069>
1050 CLOSE 1                                      <045>
```

© 64'er

Listing 2. »BIGCOPY Erzeuger«. Bitte beachten Sie bei der Eingabe den Checksummer 64 in diesem Heft.

```
100 REM  BASICVERSION DER DRUCKROUTINE            <126>
110 REM  KANN AUCH ALS UNTERROUTINE IN EIG
    ENE PROGRAMME EINGEBAUT WERDEN                <085>
120 REM  ANPASSUNG AN ANDERE 7-NADEL-DRUCK
    ER SICHER MOEGLICH                            <214>
130 OPEN 4,4:OPEN 6,4,6:PRINT#6,CHR$(16):C
    LOSE 6:POKE 56,30                             <183>
140 REM  OFFNET DRUCKKANAL UND STELLT ZEIL
    ENABSTANT FUER GRAFIK EIN                     <103>
150 FOR I=319 TO-3 STEP -7                        <019>
160 U$="":U2$="":O$="":O2$=""                     <109>
170 FOR Y=0 TO 199:C=128:G=128                    <087>
180 FOR X=I TO I-2 STEP -1                        <194>
190 A=0:GOSUB 1000                                <237>
200 BI=2*(I-X)                                    <021>
210 IF A=1 THEN C=C+2↑BI+2↑(BI+1)                 <059>
220 NEXT X                                        <168>
230 A=0:X=I-3:GOSUB 1000                          <200>
240 IF A=1 THEN C=C+64 :G=G+1                     <170>
250 FOR X=I-4 TO I-6 STEP -1                      <010>
260 A=0:GOSUB 1000                                <053>
270 BI =2*(I-X-4)                                 <217>
280 IF A=1 THEN G=G+2↑(BI+1)+2↑(BI+2)             <164>
290 NEXT X                                        <240>
300 IF Y<100 THEN O$=O$+CHR$(C)+CHR$(C)           <166>
310 IF Y>99 THEN O2$=O2$+CHR$(C)+CHR$(C)          <230>
320 IF Y<100 THEN U$=U$+CHR$(G)+CHR$(G)           <048>
330 IF Y>99 THEN U2$=U2$+CHR$(G)+CHR$(G)          <115>
340 NEXT Y                                        <042>
350 PRINT#4,CHR$(8)CHR$(27)CHR$(16)CHR$(0)
    CHR$(40);                                     <073>
360 PRINT#4,O$O2$CHR$(9)                          <090>
370 REM  CHR$(8) STELLT BIT-MUSTER-MODUS E
    IN                                            <229>
380 REM  CHR$(27)CHR$(16)CHR$(0)CHR$(40) S
    TELLT DIE DRUCKSTARTADRSSE JEDER              <185>
390 REM  ZEILE AUF 40 PUNKTE NACH RECHTS E
    IN                                            <013>
400 REM CHR$(9) STELLT DEN BIT-MUSTER-MODU
    S WIEDER AUS                                  <114>
410 PRINT#4,CHR$(8)CHR$(27)CHR$(16)CHR$(0)
    CHR$(40);                                     <133>
420 PRINT#4,U$U2$CHR$(9)                          <090>
430 NEXT I                                        <004>
440 CLOSE 4:END                                   <144>
450 REM  PRUEFROUTINE OB DER GRAFIKPUNKT G
    ESETZT ODER NICHT GESETZT IST:                <047>
1000 IF X<0 THEN A=0:RETURN                       <022>
1010 BY=(X AND 504)+40*(Y AND 248)+(Y AND
     7)                                           <029>
1020 BI=2↑(7-(X AND 7))                           <051>
1030 S=PEEK(8192+BY)                              <218>
1040 IF(S OR BI)=S THEN A=1                       <212>
1050 RETURN                                       <092>
```

© 64'er

Listing 3. »Basic Druck«. Bitte beachten Sie bei der Eingabe den Checksummer 64 in diesem Heft.


```
600 INPUT"[CLR,DOWN]NUMMER DES BILDES";D:         <129>
610 FOR C=1 TO 7:Z=PEEK(8085+C)                   <250>
620 IF Z=D THEN G=32*D:GOTO 650                   <145>
640 NEXT                                          <142>
650 IF A=5 THEN A=4:LOAD"HI-PRINT",8,1            <154>
660 POKE 3340,G:POKE 3336,32:SYS 3328:POKE
    53265,59:POKE 53272,24                        <076>
665 SYS 3383                                      <182>
670 POKE 3340,32:POKE 3336,G:SYS 3328:POKE
    53265,155:POKE 53272,21                       <091>
680 IF A=5 THEN RETURN                            <229>
690 A=1 GOTO 50                                   <249>
```

© 64'er

Listing 4. »Anp. Hi-Eddi«. Bitte beachten Sie bei der Eingabe den Checksummer 64 in diesem Heft.

```
10 REM  ZUR ERZEUGUNG VON HI-PRINT                <032>
20 REM  BIGCOPY DATA PROGRAMM EINLADEN,           <245>
30 REM  ZEILE450-1000 LOESCHEN                    <178>
40 REM  DAFUER FOLGENDE ZEILEN ANHAENGEN          <241>
450 OPEN 1,8,1,"HI-PRINT"                         <164>
460 PRINT#1,CHR$(0);CHR$(13);                     <244>
470 FOR T=1 TO 890:READ A                         <231>
480 PRINT#1,CHR$(A);                              <208>
490 NEXT                                          <246>
500 CLOSE 1                                       <001>
```

© 64'er

Listing 5. »Hi-Print Erzeuger«. Bitte beachten Sie bei der Eingabe den Checksummer 64 in diesem Heft.

Ergänzen Sie jetzt Ihre 64'er-Sammlung!

ACHTUNG: Die Ausgaben 5 und 6/84 sind vergriffen und nicht mehr lieferbar!

Schaffen Sie sich ein interessantes Nachschlagewerk und gleichzeitig ein wertvolles Archiv!

Kennen Sie alle Ausgaben von 64'er? Suchen Sie einen ganz bestimmten Testbericht? Oder haben Sie einen Teil eines interessanten Kurses versäumt? Suchen Sie nach einer speziellen Anwendung?

Damit Sie jetzt fehlende Hefte mit »Ihrem« Artikel nachbestellen können, finden Sie auf diesen Seiten eine Zusammenstellung aller wesentlichen Artikel von Ausgabe 4/84 bis Ausgabe 3/85.

Und so kommen Sie schnell an die noch lieferbaren Ausgaben: Prüfen Sie, welche Ausgabe in Ihrer Sammlung noch fehlt, oder welches Thema Sie interessiert. Tragen Sie die Nummer dieser Ausgabe und das Erscheinungsjahr (z.B. 2/85) auf dem Bestellabschnitt der hier eingehefteten Bestell-Zahlkarte ein. Die ausgefüllte Zahlkarte einfach heraustrennen und Rechnungsbetrag beim nächsten Postamt einzahlen. Ihre Bestellung wird nach Zahlungseingang umgehend zur Auslieferung gebracht.

Aktuell

Stichwort	Titel	Seite	Ausgabe
Computer	Die neuen — 264 und 364 (von der CES in Las Vegas)	9	4/84
	Heiße Messe in der Wüste: CES (PC 128, PC 10, Commodore LCD)	8	3/85
DFÜ	Datex-P und ausländische Netzwerke	59	10/84
	Informationen zur Datenfernübertragung	8	11/84
	Interessant bis brisant — die elektronischen Briefkästen	10	12/84
	Internationaler Chaos Communiction Congress	15	3/85
	Kreatives Chaos (Interview mit dem CCC)	12	10/84
	MCI Mail: die schnelle Post	8	2/85
Floppy	Neues 1451-Laufwerk	14	3/85
	SFD 1002	8	9/84
Messen	Commodore Fachaustellung in Frankfurt	10	11/84
	Consumer Electronics Show in Chicago	10	8/84
	Die Neuigkeiten von der Hannover Messe '84	8	6/84
Musik	Musikneuigkeiten aus den USA — MIDI	44	9/84
Raubkopierer	Der Sumpf	12	6/84

Listings zum Abtippen

Anwendung

Stichwort	Titel	Seite	Ausgabe
Abtippen	Checksummer (C 64 und VC 20)	72	1/85
	Checksummer (C 64 und VC 20)	65	2/85
	MSE — Abtippen sicher und leicht gemacht	68	2/85
	MSE — Abtippen sicher und leicht gemacht	78	3/85
	Neuer Checksummer 64 — blitzschnell und kürzer	68	3/85
DFÜ	Mailboxprogramm für den C 64	114	9/84
	Terminalprogramm für den C 64	24	7/84
Datei	Adreß- und Telefonregister	64	5/84
	Relative Programm-Datei (VC 20)	69	5/84
EPROM	Datenbrennerei: Wie programmiere ich EPROMs?	162	9/84
Familie	Familienplanung mit dem VC 20 (AdM)	52	2/85
Finanzen	Abgerechnet wird mit dem C 64 (AdM)	68	8/84
	Menügesteuerte Finanzmathematik (AdM)	68	10/84
Floppy	Der Softwarekatalog für Ihre Programme	72	7/84
	Drucker/Floppy ein- oder ausgeschaltet?	77	8/84
	Hypra-Load: Schnelles Laden von Diskette (LdM)	67	10/84
Kalender	Elekronisches Notizbuch (VC 20)	50	4/84
Lehrer	Lehrerkalender (AdM)	64	6/84
Lernprogramme	Russische Vokabeln (nur auf Disk)	76	7/84
Maske	Bildschirmasken schnell erstellt	78	9/84
Mathematik	Mathematical-Basic: Das Super-Basic für den VC 20 (LdM)	50	12/84
Monitor	Ohne gutes Werkzeug geht es nicht: SMON (Teil 1)	60	11/84
	Ohne gutes Werkzeug geht es nicht: SMON (Teil 2)	61	12/84
	Ohne gutes Werkzeug geht es nicht: SMON (Teil 3)	69	1/85
	Ohne gutes Werkzeug geht es nicht: SMON (Teil 4)	72	2/85
Morsen	Kurz - lang - kurz: kein Problem	72	6/84
Musik	Die Musik macht der C 64: Elektronikorgel (AdM)	70	9/84
	Musik, Musik, Musik: Synthesizer (AdM)	51	12/84
Schach	Schachmeister (AdM)	50	11/84
Sport	Computer und Sport — Auswertung von DMM-Ereignissen	56	4/84
	Der C 64 als Handballtrainer (AdM)	52	1/85
	Gut Ziel mit dem C 64 (AdM)	52	3/85
	Ohne Organistaion kein Tor: Ligatab (LdM)	50	3/85
Strubs	Das erste Strubs-Listing (VC 20)	65	7/84
Super 8	VC 20 steuert Super 8-Kamera	70	2/85
Textverarbeitung	Quicktext — Textverarbeitung zum Abtippen	60	6/84
User-Port	Analoger Meßwert rein — analoger Stellwert raus	78	8/84
	Blumen gießen mit dem C 64 (AdM)	82	7/84
	Kopplung über den User-Port (VC 20)	73	8/84
Video	Video-Vorspann mit dem VC 20	80	10/84
Vokabeln	Supervoc	69	6/84
Zeichensatz	Deutscher Zeichensatz für den VC 20	79	9/84

Grafik

Stichwort	Titel	Seite	Ausgabe
Algorithmus	Ein schneller Drawline-Algorithmus	65	4/84
Auflösung	Supergrafik II (VC 20)	73	11/84
	Supergrafik ohne Erweiterungsmodul	81	5/84
Axiometrie	Von allen Seiten betrachtet (SB)	69	12/84
Befehlserweiterung	Screen Change	94	9/84
Diagramme	Grafische Darstellung statt Zahlenfriedhof (SB)	82	6/84
Elektrotechnik	Elektrotechnisches Zeichnen mit dem VC 20	71	3/85
Funktionen	Grafik leicht gemacht	70	11/84
	Kudiplo erfüllt Schülerträume (Kurvendiskussion auf dem VC 20)	80	8/84
	Kurvendiskussion in Hires-Grafik mit Hardcopy	116	7/84
Grafik	Bewegte Grafik und Text mischen	66	12/84
Hardcopy	1520-Hardcopy mit dem VC 20	87	9/84
	Der VC 1525/MPS 802 als Grafikdrucker	83	10/84
	Die mehrfarbige Hardcopy mit dem 1520-Plotter	84	10/84
	Ein eigentlich unmögliches Programm (1526)	74	5/84
	Hardcopy Epson FX-80	88	10/84
	Hardcopy Gemini-10X	85	10/84
	Hardcopy MPS 801/VC 1515	82	10/84
	Hardcopy für den Sieger (FX-80 mit Gör[illegible]face)	83	8/84
	Hardcopy mit dem VC 1520	108	7/84
Joystick	3D-Grafik mit dem Joystick (VC 20)	78	8/84
KoalaPad	Get Koala Pic	66	11/84
Schnittstellen	Olympia compact 2: ein Centronics-Interface	86	10/84
Sprites	Der Super-Sprite-Editor	89	9/84
	Pseudo-Sprites auf dem VC 20	76	11/84
	Sprites ohne Esoterik	74	11/84
	Sprites schneller bewegen	70	4/84
	Vier Pseudo-VICs mit 32 Sprites	76	1/85
Trickfilm	Der VC 20 als Laterna Magica	68	11/84
	Movemaster (VC 20)	78	6/84
Turtle-Grafik	Turtle-Grafik — Die schnelle Schildkröte (LdM)	48	11/84
Zeichnen	HI-EDDI: Ein fantastisches Zeichen und Malprogramm (LDM)	50	1/85

Spiel

Stichwort	Titel	Seite	Ausgabe
Abenteuer	Castle of Doom — Adventure (LdM)	66	8/84
	Das Grab des Pharao (LdM)	51	2/85
Action	Apocalypse now	106	10/84
	Croussaider (VC 20)	124	7/84
	Ghost Manor (VC 20)	104	6/84
	Q + Bert (VC 20)	78	2/85
	Q-Bernd (LdM)	142	7/84
	Schatzsucher (LdM)	90	5/84
Arcade	Invaders	74	4/84
	Space Invader (nur auf Disk)	81	7/84
Denkspiel	3D-Vier gewinnt — Spielen in der dritten Dimension	96	12/84
	Mastermind als Vierzeiler	81	12/84
	Rätsel — ein Knobelprogramm (VC 20)	122	7/84
Generator	Spring Vogel, spring (LdM)	68	9/84
Geschicklichkeit	In den Untergrund mit dem VC 20 (LdM)	120	6/84
Pacman	Pac-Boy — die Herausforderung	89	8/84
	Schmatzer (VC 20)	76	5/84
Reaktion	Escape (VC 20)	85	8/84
Simulation	Fahrsimulator	82	5/84
	Hot Wheels	98	6/84
	Rennfahrer ohne Sturzhelm (VC 20)	86	4/84
Spielhalle	Crown No 1 (nur auf Disk)	80	7/84
Strategie	Schiebung (VC 20)	77	9/84
Taktik	Epedemic (VC 20)	112	10/84
	Gehirntraining mit Supermemory	81	2/85
	Kämpfe wie im alten Rom — Caesar	78	4/84
Wettbewerb	Notlandung	156	2/85

Tips & Tricks

Stichwort	Titel	Seite	Ausgabe
Absturz	Die Ebenen des Absturzes	92	11/84
Auto	Automatische Zeilennumerierung	84	12/84
Autoboot	Autoboot beim C 64	86	3/85
Autostart	Autostart für den VC 20	98	8/84
	Autostart in Theorie und Praxis	138	7/84
	Der Trick mit dem Autostart	117	6/84
Basic	Basic-Programme stutzen (VC 20)	85	5/84
	Basic-Zeilen genau betrachtet	87	2/85
Basic-Erweiterung	PRINT AT und RESTORE N (VC 20)	101	8/84
	Stringy: C 64-Erweiterung	86	12/84
Buchstaben	Große Buchstaben	89	1/85
Datasette	Fast Tape — die schnelle Kassette (VC 20)	80	12/84
	Musik aus der Datasette	84	12/84
Direktmodus	Programmierter Direktmodus	82	12/84
Disassembler	Logic Disassembler (VC 20)	108	6/84
Erweiterung	Betriebssystem-Erweiterung für den VC 20	88	11/84
FRE	Veränderungen am 64-Basic	119	6/84
Floppy	22 Read Error — Theorie und Praxis	41	3/85
	Auf das "!" kommt es an	92	12/84
	Disk Copy	92	4/84
	Diskette intern (Disk-Dump)	95	10/84
	Disketten-Organisation (VC 20)	97	10/84
	Diskettenzauberei	88	5/84
	Floppy-Lister	82	3/85
	Hypra-Load mal vier	82	1/85
	Kopieren mit Komfort: Super Copy	102	10/84
	Kopierprogramm für relative Files	132	7/84
	Maschinenprogramme auf Diskette speichern	91	2/85
	View BAM	99	8/84
Funktionen	Kudiplo auf für den C 64 (Kurvendiskussion)	91	10/84
Generator	Dem Springvogel auf die Sprünge geholfen	82	11/84
Grafik	Tips und Erweiterungen zu Hi-Eddi und Simons Basic	88	3/85
	Vom Bildschirm auf Kassette/Diskette	135	7/84
Hex-Data	Hex-DATA-Automat	81	11/84
Interrupt	Unterbrechen Sie mich bitte!	84	11/84
Joystick	Cursorsteuerung leicht gemacht (mit Joystick)	86	2/85
	Joystick-Abfrage in Theorie und Praxis	83	11/84
Listing	Der große Überblick: formatiertes Listing	90	10/84
	Fehlersuche leicht gemacht: LIST-STOP	97	9/84
	Programmiertes LISTing: LIST X-Y	100	10/84
Listschutz	List- und Löschschutz leicht gemacht	85	12/84
Maschinensprache	Maschinenprogramme auf Tastendruck	80	12/84
	DATA-Wandler	102	9/84
Merge	Kleben per Software — Merge	94	4/84
Module	Steckmodule auf Kassette (VC 20)	107	6/84
Monitor	Besseres Monitorbild beim C 64	90	2/85
Opcodes	Hex-ereien: undefinierte Opcodes beim 6502	84	3/85
POKEs	Durch POKEs zum Erfolg — Die Spiele-Trickkiste	83	3/85
	POKE mal wieder: diverse POKEs	91	10/84
Parameter	Parameterübergabe an Programme in Maschinensprache	88	1/85
Reset	Reset für den VC 20	116	6/84
	Resetschalter am C 64	34	8/84
Restore	Restore für Unterprogramme	90	1/85
Retten	Erste Hilfe (VC 20)	88	4/84
	Erste Hilfe für den C 64: RENEW	102	9/84
SYS	Mehr über SYS	131	7/84
Schnittstellen	Die RS232-Schnittstelle am VC 20	100	9/84
	Komfortables Treiberprogramm für Centronics-Drucker	110	7/84
	Verbindungsfreundlich (VC 20)	91	3/85
Scrollen	Als die Bilder laufen lernten ... (Scrollen)	88	2/85
Simons Basic	Befehlserweiterung für Simons Basic	90	11/84
	Haben Sie den Bogen raus? (ARC bei Simons Basic)	98	9/84
	Simons Basic: Befehle die nicht im Handbuch stehen	103	9/84
Speicher	RAM-Floppy	92	2/85
Synthetische	Die Suche nach den Synthetischen	104	9/84
	Synthetische Steuerzeichen	114	6/84
	Synthetische Steuerzeichen (Teil 2)	136	7/84
Tastatur	User-Port-Tastatur (Zehnertastatur)	93	10/84
Tips & Tricks	Diverse	89	10/84
	Diverse Tips & Tricks für den C 64	86	5/84
	Lösung von Dallas Quest	90	1/85
	Tips & Tricks für den C 64	108	4/84
	Tips & Tricks für den VC 20	118	6/84
Trace	Trace und Single Step für Maschinenprogramme	76	12/84
Transfer	Adressenvergleich VC 20-C 64	52	7/84
	Der C 64 als PET	87	1/85
	Die Software-Vielfalt der CBMs für den C 64 nutzen	102	8/84
	Von den Kleinen auf die Großen (C 64 - CBM)	96	8/84
User-Port	User-Port-Display	97	8/84
	User-Port-Tastatur (Zehnertastatur)	93	10/84

Hardware-Test

Stichwort	Titel	Seite	Ausgabe
80-Zeichenkarten	Mehr Übersicht am Bildschirm (VC 20)	20	10/84
Computer	C 16: Der Nachfolger?	20	6/84
	Ein Wolf im Schafspelz — der 264 (Plus/4)	14	5/84
	Generationswechsel — Test C 16	6	1/85
	Plus und Minus beim Plus/4	14	2/85
	SX 64 im Test	27	4/84
Drucker	Adcomp X100 — farbig plotten und drucken	22	10/84
	Brother HR-5C: fast nicht zu hören	24	10/84
	Commodore-Drucker unter 1000 Mark	30	4/84
	Der Petal MA20 — kleiner Name, großer Drucker	24	11/84
	Die farbig druckende Kreissäge GP-700A	24	5/84
	Drucksympathie (BMC BX100)	26	11/84
	Ein Drucker für alle Fälle: Epson FX-80	23	10/84
	Ein Star der es in sich hat (delta-10)	25	10/84
	Ein starkes Stück (Itoh 8510)	22	11/84
	Olympia electronic compact 2: Schreibmaschine für den C 64	28	10/84
	Roland DXY-101 — ein Flachbettplotter im DIN-A3-Format	27	10/84
	Seikoshas Größter: Test GP-550A	26	10/84
EPROM	EPROM-Brenner: Vergleichstest	36	8/84
Erweiterungen	KFC-Super — ein Super-Toolkit?	20	5/84
Expansions	Expansions über alle Grenzen hinaus	34	4/84
Floppy	Floppy mit Nachbrenner (Speeddos, Turboaccess, Computing)	26	12/84
	MSD-Super-Disk-Drive	14	11/84
Grafik	Grafiktabletts: KoalaPad und SuperSketch	34	11/84
Interface	Das macht den Kleinen größer (64-KByte-Karten) (VC 20)	112	9/84
	Digitalisierte Bilder mit dem C 64	24	1/85
	Speichertuning für VC 20: MR 64	26	1/85
Joystick	Joystick-Vielfalt (20 Joysticks im Test)	34	12/84
Monitore	Die Scharfmacher (Cable, Taxan, BMC)	20	12/84
	Monitor kontra Fernseher (Taxan, 1701, Zenith, Sharp, Panasonic)	16	11/84
Schnittstellen	Card/Print + G — Das Allround-Interface (Centronics)	20	3/85
	Centronics-Schnittstellen: Vergleichstest	18	7/84
	Das Interface mit Weitblick (WW-92000/G: Centronics)	18	3/85
	Hardware-Interface ganz weich: EC-64	23	1/85
	Print 64 — das universelle Interface (Centronics)	24	12/84
Sprachausgabe	Die Stimme des Meisters: Voice Master	19	2/85
	Sprachausgabe mit dem SDP 120	22	8/84

Hardware

Stichwort	Titel	Seite	Ausgabe
Bauanleitung	16-KByte-Erweiterung umschaltbar (VC 20)	20	2/85
	Commodore im neuen Kleid	30	8/84
	Das 30-Mark-Interface (RS232)	29	3/85
	Ihr Akustikkoppler wird zum Modem: Automodem	114	9/84
	Joystick im Selbstbau	33	3/85
	Resetschalter am C 64	34	8/84
	Richtig verbunden — Video/Audio-Kabel für den C 64	22	2/85
DFÜ	Akustikkoppler und Modems: Marktübersicht	28	8/84
Datasette	Rund um die Datasette	30	6/84
Drucker	MPS 801 — Ein Erfahrungsbericht	20	8/84
	Marktübersicht: Drucker (Teil 1)	29	10/84
	Marktübersicht: Drucker (Teil 2)	21	11/84
EPROM	Nichts ist ewig (ROM-Change, verbessertes Betriebssytem)	30	12/84
Expansion	Expansions — Marktübersicht	18	7/84
Grundlagen	Der serielle Bus des VC 20 und C 64	28	5/84
Monitor	Richtig verbunden — Video/Audio-Kabel für den C 64	22	2/85
	Marktübersicht: Schwarzweiß- und Farbmonitore	19	11/84
Musik	MIDI — Glanz und Elend eines Interfaces	46	9/84
Reparatur	Geheimnissen auf der Spur: 1541 reparieren	24	8/84
Schnittstellen	Centronics-Schnittstelle für Epson FX-80 (mit Kabel)	34	6/84
	Erst ein IEC-Bus öffnet Tür und Tor (Marktübersicht und Test)	24	3/85
	Gute Connections (RS232-, Centronics-Marktübersicht)	21	3/85
	Kopplung zwischen zwei VC 20	24	6/84

Alle Beiträge sind in der Regel für den C 64, sofern nicht anders gekennzeichnet (VC 20).
Folgende Abkürzungen wurden verwendet: LDM = Listings des Monats, AdM = Anwendung des Monats, SB = Simons Basic.

Jetzt zugreifen, solange ältere Ausgaben und Sonderhefte noch lieferbar sind!

Auch die bisher erschienen Sonderhefte können Sie jetzt direkt bestellen:

TIPS & TRICKS
(1. Programm-Sonderheft)
Eine wahre Fundgrube an Ideen und Programmen für Computer-Profis und alle, die es werden wollen.
BESTELLCODE: Tips&Tips

ABENTEUERSPIELE
(2. Programm-Sonderheft)
Auf mehr als 100 Seiten viele interessante Adventures, die Sie lange Zeit fesseln werden. Mit abgeschlossenem Kurs zur Programmierung eigener Abenteuerspiele und zahlreichen Lösungen professioneller Adventures.
BESTELLCODE: Abenteuerspiele

SPIELE
(3. Programm-Sonderheft)
Heiße Listings für alle Spiele-Fans: Sportspiele, Schießspiele, Denkspiele, Spielegeneratoren, Abenteuerspiele, Brettspiele, Taktikspiele, Geschicklichkeitsspiele und eine Marktübersicht aller in Deutschland erhältlichen professionellen Spiele bringen alles, was das Herz der Spiele-Fans höher schlagen läßt.
BESTELLCODE: Spiele

Neu: die 64'er-Sammelbox!

Für alle Leser, die »64'er« regelmäßig kaufen, sammeln oder im Abonnement beziehen, gibt es jetzt ein interessantes Service-Angebot: die 64'er-Sammelbox!

Mit dieser Sammelbox bringen Sie nicht nur Ordnung in Ihre wertvollen Hefte, sondern schaffen sich gleichzeitig ein interessantes und attraktives Nachschlagewerk.

Übrigens: Die Sammelbox ist nicht nur ein praktisches Aufbewahrungsmittel: Sie eignet sich auch hervorragend als Geschenk für Freunde und Bekannte zu vielen Anlässen.

Ein kompletter Jahrgang (12 Hefte) paßt in die praktische Sammel-Box! Am besten gleich bestellen!

Hardcopy mit dem Seikosha GP-550

Über den seriellen Bus des C 64 können mit dieser Routine durch einen SYS-Befehl Grafiken in einfacher und in doppelter Größe zu Papier gebracht werden.

Tippen Sie zunächst den Basic-Lader (Listing 1) ab und speichern ihn auf Diskette. Anschließend ist das Programm mit RUN <Return> zu starten. Es wird ein Maschinenprogramm ab Speicherzelle 49152 erzeugt und nach korrekter DATA-Summenkontrolle unter dem Namen »M/HCOPY GP-550A« abgespeichert. Tritt bei der DATA-Summenkontrolle kein Fehler auf, kann das Maschinenprogramm wie folgt aufgerufen werden:

Syntax der Hardcopyroutine:
SYS49152⟨(Pg)⟩⟨,M,(L-1)*4+Br,LM⟩⟨;⟩

Hierbei bedeuten:

⟨...⟩ Ein in spitzen Klammern stehender Ausdruck ist optional, kann also weggelassen werden.

Pg Grafikseitennummer 0 < Pg < = 7.
Durch die Angabe dieser Grafikspeicherkennziffer wird grundsätzlich der RAM-Bereich angesprochen!
Zum Beispiel:
(0): nicht erlaubt (illegal quantity error), da dieser Bereich die Zeropage beinhaltet.
(1): Standardgrafikspeicher (8192 bis 16383); wird auch angesprochen, wenn keine Angabe zu Pg gemacht worden ist.
(5): spricht das RAM unter dem Basic-ROM an.
(7): Hardcopy einer Simons-Basic-Grafik, die unter dem Kernal-ROM liegt.

M Moduskennzahl M = 0 (128) : 8-bit normal (revers)
M = 1 (129) : 16-bit normal (revers)

L Länge L = 1 : einfache Länge / L = 2 : doppelte Länge
Beachten Sie, daß die Zahl (L-1) einzugeben ist.

Br Breite 1 < = Br < = 3

LM Linker Randabstand (siehe Druckeranleitung)
Der größtmögliche Wert des Randabstandes ist von der Breite abhängig. Eine Überschreitung des erlaubten Wertes ergibt jedoch »illegal quantity error«, so daß kein Fehlausdruck durch Zeilenüberlauf entstehen kann.
Zusätzlich stellt Ihnen die Routine eine Zentrierungsautomatik zur Verfügung, die Sie durch »LM=1« aktivieren können.
sperrt Blattvorschub nach erfolgtem Ausdruck, so daß Sie mehrere Hardcopies abstandsfrei koppeln können.

Beispiele:
(1) **SYS49152**
Ausdruck der Standard-Bitmap (8192-16383) mit Normalbelegung der Parameter. (8-Bit normale, doppelte Breite, einfache Länge, mittenzentriert, Seitenvorschub)
(2) **SYS49152(7);**
Hardcopy einer Simons Basic-Grafik wie oben, aber ohne Seitenvorschub.
(3) **SYS49152,1,1,0**
Kleinstformat in 16-Bit-Auflösung ohne Randabstand.
(4) **SYS49152(3),128,6,5;**
Darstellung der Bitmap von 24576 bis 32767 im 8-Bit-Revers-Modus. Die Breite beträgt wie die Länge zwei, der linke Rand fünf Einheiten.

Um dem Anwender eine möglichst flexible Nutzungsmöglichkeit anbieten zu können, stehen neben der üblichen, oben dargestellten Parameterübergabe zusätzliche POKE-Optionen zur Verfügung, die zum Beispiel die Rahmenerstellung oder das Druckbild beeinflussen.

Alle Möglichkeiten sind in den Tabellen 1 und 2 aufgeführt. Wichtig dabei ist:

Sämtliche in Tabelle 1 aufgelisteten POKE-Optionen sind nur für die aktuelle Hardcopy gültig und werden nach Beendigung des Druckvorganges automatisch abgeschaltet!

Hingegen sind die Programmeingriffe, die in Tabelle 2 stehen, dauerhaft, und müssen explizit zurückgesetzt werden.

Adresse	Wert	
49686	96	Rahmen ausschalten
49741	0	oberen ... Rahmenteil weglassen
49731	0	unteren ... Rahmenteil weglassen
49705	11	linken ... Rahmenteil weglassen
49699	11	rechten ... Rahmenteil weglassen
49515	0	Druckbild auflockern (M=0, BR=2)
49531	96	Steuerung der Zeilenausgabe in Basic ermöglichen
Einsprungadressen:		
49532	(*)	nächste Grafikzeile bearbeiten; Fortsetzung: mit CHR$ (13)
49537	(*)	nächste Grafikzeile bearbeiten; Fortsetzung: ohne CHR$ (13)
49398	(**)	alternativen Grafikbereich in Druckerpuffer bringen (siehe Listing 3)

(*) Im Unterschied zu (**) werden zunächst interne Variablen angepaßt und dann nach (**) verzweigt.

Tabelle 1. Einmalig wirksame POKE-Optionen sowie Einsprungadressen

Die Option **POKE49531,96** trägt wesentlich dazu bei, die Hardcopyroutine um interessante Aspekte zu bereichern. Sie sorgt dafür, daß jedesmal, wenn eine Grafikzeile abgearbeitet worden ist, die Kontrolle zwischenzeitlich an Basic zurückgegeben wird.

Dies eröffnet die Möglichkeit, vor oder hinter jeder Grafikzeile einen Text zu senden. Mit einem kleinen Trick sind zwei oder drei Grafikseiten nebeneinander zu erstellen, vorausgesetzt, die maximale Zeichenanzahl wird nicht überschritten.

Die prinzipielle Vorgehensweise dabei zeigen die Listings 2 und 3. Dort verwendete Adressen finden sich auch in Tabelle (1).

Listing 3 läßt sich noch effektvoll erweitern, indem geschickt der Elongation-Modus durch »PRINTCHR$(14);« zum Beispiel in Zeile 240 beziehungsweise »...CHR$(15);« in Zeile 210 ein- oder ausgeschaltet wird. Man erhält so zwei nebeneinander gedruckte Hardcopies mit einfacher und doppelter Breite.

Adresse	beeinflußt	Funktion
49260	Geräteadresse	Anpassen an zweiten Drucker
49262	Sekundäradresse	Einschalten des Groß-/Kleinschrift-modus des Druckers (7); Anwendung wie Listing 1.
49764	LF-Spacing (siehe Drucker-anleitung)	Zeilenabstand verringern (56); Komprimierte Darstellung
49384	Zeilenanzahl	Grafikbildschirm nur teilweise reproduzieren.
49399	Spaltenanzahl	

Tabelle 2. Adressen für dauerhafte Veränderungen durch POKE-Befehle

In diesem Zusammenhang sei noch erwähnt, daß nach dem ersten Aufruf der Hardcopy-Routine alle Druckerbefehle nur noch mit einem normalen PRINT-Befehl gesendet werden dürfen, solange der Ausdruck nicht vollständig abgeschlossen ist.

(Frank Jörgens/ah)

```
 programm : hcopy gp550       c000 c275
 ----------------------------------------

c000 : 20 79 00 c9 28 d0 18 20  c0
c008 : f1 ae 20 a1 b7 ca 30 56  cc
c010 : f0 0d ca e0 06 b0 4f 48  09
c018 : bd 6f c2 8d e1 c0 68 c9  48
c020 : 2c f0 04 a2 01 d0 35 20  c5
c028 : 9b b7 8e 52 c2 a0 00 20  fe
c030 : f1 b7 8a 29 03 f0 2f 8d  54
c038 : 54 c2 8a 29 0c 4a 4a aa  47
c040 : 86 02 f0 08 ad 52 c2 29  cf
c048 : 80 8d 52 c2 e8 8e 53 c2  52
c050 : 20 f1 b7 8a a8 ae 54 c2  7f
c058 : ca 88 d0 03 bd 5b c2 85  c8
c060 : 60 dd 5e c2 90 03 4c 48  82
c068 : b2 a9 7f a2 04 a0 86 20  c3
c070 : ba ff 20 c0 ff a2 7f 20  9d
c078 : c9 ff a2 06 bd 61 c2 20  dc
c080 : dd ed ca 10 f7 a4 60 c0  b0
c088 : 0a 08 20 a2 b3 a0 00 20  73
c090 : df bd 28 b0 0b ad 00 01  8e
c098 : 8d 01 01 a9 30 8d 00 01  8d
c0a0 : ad 00 01 20 dd ed ad 01  70
c0a8 : 01 20 dd ed 20 79 00 c9  50
c0b0 : 3b d0 06 8d 55 c2 20 73  59
c0b8 : 00 a9 47 20 f2 c1 90 02  e6
c0c0 : a9 49 8d 59 c2 a9 30 aa  2c
c0c8 : ac 54 c2 18 69 03 e8 e8  76
c0d0 : 88 d0 f8 8d 58 c2 8e 57  35
c0d8 : c2 a9 00 85 d1 8d 50 c2  70
c0e0 : a9 20 85 d2 8d 51 c2 a9  17
c0e8 : 19 8d 4f c2 ac 50 c2 84  55
c0f0 : d1 ac 51 c2 84 d2 a9 28  9a
c0f8 : 85 d7 a2 04 bd 56 c2 20  6c
c100 : dd ed ca 10 f7 a2 07 a0  7a
c108 : 07 84 9b 78 a5 01 48 29  1d
c110 : fc 85 01 b1 d1 95 19 ca  09
c118 : 88 10 f8 68 85 01 a5 d1  8e
c120 : 18 69 08 85 d1 90 02 e6  17
c128 : d2 a2 07 16 19 08 2a ac  a4
c130 : 53 c2 c0 02 f0 05 20 f2  f2
c138 : c1 90 0c 28 08 2a e0 04  a7
c140 : d0 05 20 09 c2 85 09 28  89
c148 : ca 10 e0 20 09 c2 ac 53  56
c150 : c2 88 f0 07 a4 02 88 f0  d2
c158 : 02 a5 09 ac 54 c2 c0 02  67
c160 : 90 0e f0 05 48 20 fc c1  d1
c168 : 68 48 29 ff 20 fc c1 68  00
c170 : 20 fc c1 c6 9b 10 b2 c6  ea
c178 : d7 d0 8a ea a9 0d 20 dd  f6
c180 : ed ac 53 c2 88 f0 07 c6  aa
c188 : 02 d0 03 4c ec c0 84 02  28
c190 : a4 d1 8c 50 c2 a4 d2 8c  00
c198 : 51 c2 ce 4f c2 f0 03 4c  40
c1a0 : f6 c0 ac 55 c2 d0 05 a9  e6
c1a8 : 0c 20 dd ed a2 06 bd 68  1b
c1b0 : c2 20 dd ed ca 10 f7 a9  18
c1b8 : 7f 20 c3 ff a2 ff 8e 6b  73
c1c0 : c1 e8 8e 52 c2 8e 55 c2  5f
c1c8 : e8 8e 53 c2 8e 4d c2 e8  55
c1d0 : 8e 54 c2 a2 80 8e 43 c2  9c
c1d8 : a2 07 8e 29 c2 a9 08 8d  7b
c1e0 : 23 c2 a9 20 8d e1 c0 a9  11
c1e8 : ea 8d 7b c1 8d 16 c2 4c  dd
c1f0 : cc ff ac 52 c2 4e 52 c2  9f
c1f8 : 8c 52 c2 60 20 dd ed 20  53
c200 : f2 c1 90 f7 a5 09 4c dd  85
c208 : ed ac 52 c2 0e 52 c2 8c  d0
c210 : 52 c2 90 02 49 ff ea a4  b1
c218 : d7 c0 28 f0 09 88 d0 0f  ae
c220 : a4 9b f0 08 d0 09 a4 9b  ee
c228 : c0 07 d0 03 a9 ff 60 ac  76
c230 : 4f c2 88 d0 0f 20 f2 c1  5e
c238 : 90 04 e0 04 d0 f0 a4 02  ae
c240 : d0 ec 09 80 e0 04 f0 e6  98
c248 : c0 18 d0 e2 09 01 60 00  bf
c250 : 00 00 00 01 02 00 30 34  ba
c258 : 36 47 1b 1a 0d 00 36 1b  1c
c260 : 01 30 4c 1b 39 1b 18 1b  f3
c268 : 30 30 30 4c 1b 36 1b 40  96
c270 : 60 80 a0 c0 e0 ff 00 ff  5e
```

Listing 1. Basic-Lader zur Hardcopy-Routine mit dem GP-550. Bei der Eingabe bitte den MSE beachten.

```
100 REM *****************************           <091>
110 REM *     BEISPIEL FUER DIE       *          <225>
120 REM *      KOMBINATION VON        *          <114>
130 REM *    GRAPHIK UND SCHRIFT      *          <166>
140 REM *     IN EINER HARDCOPY       *          <133>
150 REM * MIT DEM SEIKOSHA GP-550A *             <003>
160 REM *****************************           <151>
170 REM                                          <232>
180 DIM T$(25)                                   <006>
185 LMAX=25:REM MAXIMALE TEXTLAENGE PRO ZE
    ILE (HIER BEISPIELHAFT 25)                   <050>
190 REM  TEXT ZEILENWEISE ABFRAGEN               <170>
200 PRINT"{CLR}";:FOR T=0 TO 24                  <184>
210 PRINT T+1 CHR$(157);                         <034>
220 INPUT". ZEILE ";T$(T)                        <130>
230 IF LEN(T$(T))>LMAX THEN PRINT"{2UP}":G
    OTO 210                                      <012>
240 T$(T)="{5SPACE}"+T$(T): REM ABSTAND SC
    HAFFEN                                       <239>
245 NEXT T                                       <161>
250 PRINT"{CLR,3DOWN}HARDCOPY ERFOLGT !"         <153>
260 REM ZEILENOPTION EINSCHALTEN                 <223>
270 POKE 49531,96                                <034>
280 REM HARDCOPY STARTEN                         <164>
```

Listing 2. Mischen von Grafik und Textausdruck

```
290 SYS 49152(1),0,1,0;:REM BEISPIELSWEISE <242>
300 REM ZEILEN EINZELN AUS GRAPHIK UND TEX
    T ZUSAMMENSETZEN                       <130>
310 FOR I=1 TO 25                          <080>
320 PRINT T$(I-1):REM TEXT UND CHR$(13)    <014>
330 SYS 49537:REM NAECHSTE GRAPHIKZEILE BE
    RECHNEN                                <193>
340 NEXT I                                 <170>
350 END                                    <098>
```


Listing 2. Mischen von Grafik und Textausdruck (Schluß)

```
100 REM ******************************** <238>
110 REM * BEISP.FUER PARALELLAUSDRUCK * <240>
120 REM *     ZWEIER GRAPHIKSEITEN    * <220>
130 REM * MIT DEM SEIKOSHA GP-550AVC  * <165>
140 REM * (ERWEITERBAR AUF DREI       * <022>
150 REM *        GRAFIKSEITEN         * <108>
160 REM ******************************** <042>
165 :                                    <141>
170 POKE 49531,96 : REM ZEILENOPTION EINSC
    HALTEN                               <015>
180 PG=7:M3=0 : M4 = PG*2*16 : REM PG = GR
    FAHIKSEITENKENNZIFFER                <076>
185 SYS 49152,0,1,0 : REM HARDCOPY STARTEN <249>
190 FOR Z=1 TO 25 : REM 25 ZEILEN (EINFACH
    E LAENGE)                            <162>
200 REM ZEIGER AUF DIE ERSTE GRAPHIKSEITE
    ZWISCHENSPEICHERN                    <212>
210 M1=PEEK(209) : M2=PEEK(210)          <215>
220 REM AUF ZWEITE GRAPHIKSEITE UMSCHALTEN <255>
230 POKE 209,M3 : POKE 210,M4            <068>
240 SYS 49398 : REM HARDCOPY-EINSPRUNGADRE
    SSE                                  <158>
250 REM ZEIGER AUF ZWEITE GRAPHIKSEITE ZWI
    SCHENSPEICHERN                       <102>
260 M3 = PEEK(209) : M4 = PEEK(210)      <091>
270 POKE 209,M1 : POKE 210,M2 : REM AUF ER
    STE GR.SEITE ZURUECK                 <174>
280 SYS 49532 : REM AUSDRUCKEN           <182>
290 NEXT Z                               <000>
300 END                                  <048>
```


Listing 3. Parallelausdruck zweier Grafiken

Ein Zeichen-generator für den FX-80

Das Programm MAKE-CHR gestattet es auf einfache Weise, komplette Zeichensätze für Drucker wie den FX-80 selbst zu erstellen.

Es sind beliebige grafische Symbole und Schriftarten, insbesondere auch Zeichensätze mit doppelter Schrifthöhe (über zwei Druckzeilen) machbar.

Die gewünschten Zeichen werden einfach mit Hilfe des Joysticks (und/oder Tastatur) in ein Bildschirmfenster gezeichnet und dann in einer relativen Datei (CHR-Datei) abgespeichert. Weiterhin enthält das Programm Routinen zum teilweisen oder vollständigen Kopieren der CHR-Dateien sowie zum Übertragen der Zeichensätze in das Drucker-RAM. Ein Drucker-Handling-Menü erlaubt das Setzen verschiedener Druckerparameter und das Ausdrucken der Zeichen. Bei jeder Eingabe kann mit der Taste »Pfeil nach links« ins Menü zurückgesprungen werden.

Bei jeder Eingabe wird ein Default-Wert angezeigt, der bei leerer Eingabe erhalten bleibt. Eine neue Eingabe wird automatisch zum neuen Default-Wert.

Die ins Drucker-RAM übertragenen Zeichensätze können von Textprogrammen benutzt werden.

Zeichenerklärung für nachfolgende Beschreibung:

& = Bildschirm * = Bemerkungen
Fett = Ihre Eingabe

ERSTELLE ZEICHEN

& Aus welcher Datei willst Du lesen? **Datei X**
& In welche Datei willst Du schreiben? **Datei Y**
& ASCII ? **ASCII X**

Die Zeichen werden aus der Datei X gelesen und in die Datei Y zurückgeschrieben.

Datei X und Datei Y können natürlich auch gleich sein. ASCII ist der ASCII-Code des zu bearbeitenden Zeichens.

1. Mit den CRSR-Tasten kann der Cursor im Bildschirmfenster plaziert werden. Die Taste »*« dient zum Setzen und Löschen von Punkten.

2. Zeichen können auch mit dem Joystick erstellt werden. Wird der Feuerknopf gedrückt, wird ein Punkt gesetzt (gelöscht).

* Es dürfen keine 2 Dots in horizontaler Linie nebeneinander liegen (Eigenheit des FX-80)!

Neun-DOT-Funktion:

Die Neun-DOT-Funktion bewirkt, daß das Zeichen um einen Dot versetzt nach unten ausgedruckt wird (für Unterlängen, Kommas und so weiter).

SEND CHR TO PRINTER

& Welche Datei soll übertragen werden ? **Datei X**
& von ASCII? **ASCII X**
& bis ASCII? **ASCII Y**

Durch die Eingabe von ASCII X und ASCII Y kann man erreichen, daß nur ein bestimmter Teilbereich des Zeichensatzes ins Drucker-RAM übertragen wird.

ASCII X=0 und ASCII Y=254 (Default-Werte, das heißt voreingestellte Werte) überträgt den gesamten Zeichensatz.

* Nach dem Übertragen eines Zeichensatzes wird ein automatischer Druckertest durchgeführt.

* Wird ein leerer Zeichensatz beziehungsweise ein Zeichensatz mit Lücken übertragen, so wird das Drucker-RAM von den Lücken nicht überschrieben. Lücken werden vom Programm berücksichtigt und übersprungen!

DRUCKER-HANDLING

1. COPY ROM TO RAM
   Kopiert das Drucker-ROM ins Drucker-RAM. Das Drucker-RAM hat dann den normalen Epson-Zeichensatz.
2. SELECT ROM/RAM
   Wählt Drucker-ROM oder -RAM aus.
3. SET LINE-SPACE
   Setzt den Drucker auf den gewählten Zeilenabstand. Bei doppelt hohem Zeichensatz wird automatisch der richtige Zeilenabstand eingestellt.
4. WÄHLE SCHRIFTART
   Auswahl einer der 64 Schriftarten des FX-80.
5. TEST PRINTER
   Druckt den gesamten ASCII-Code.

ERSTELLE CHR-DATEI

Formatiert eine neue CHR-Datei. Der Name der CHR-Datei darf maximal 10 Zeichen lang sein.

COPY SINGLE CHR

& Aus welcher Datei wird gelesen? **DATEI X**
& von ASCII? **ASCII X**
& bis ASCII? **ASCII Y**
& in welche Datei wird geschrieben? **DATEI Y**
& ab ASCII? **ASCII Z**

Kopiert den Bereich ASCII X bis Y aus DATEI X in die DATEI Y ab ASCII Z.

* DATEI X und DATEI Y müssen sich auf einer Diskette befinden.

COPY CHR-DATEI

Kopiert die angegebene CHR-Datei vollständig — auch auf eine andere Diskette. (Rainer Busch/ah)

```
1 GOTO 10                                      <179>
2 ************************************         <147>
3 *                                  *         <026>
4 *     RAINER BUESCH                *         <020>
8 *                                  *         <031>
9 ************************************         <154>
10 POKE 650,128 :REM TASTEN-REPEAT             <080>
20 PRINT CHR$(142);CHR$(8):REM LETTERS BLO
   CKED                                        <011>
30 POKE 53280,11:POKE 53281,12:REM FARBE       <070>
40 PRINT CHR$(144);                            <058>
50 OPEN 3,4,4                                  <182>
54 PRINT#3,CHR$(7) :REM BEEP                   <141>
60 PRINT#3,CHR$(27);"U";CHR$(1);               <235>
98 GOTO 200                                    <034>
99 :                                           <075>
200 REM WERTZUWEISUNGEN ***************        <158>
202 DIM ZA$(18) :REM READ RECORD               <045>
204 DIM ZA(18)  :REM DITO                      <039>
206 DIM RC$(256):REM COP-VARIABLE              <237>
208 A=139:REM ATTRIBUTE A                      <189>
210 PA=1348 :REM SCREENRAM                     <170>
212 PO=1 :REM RECORD POSITION                  <061>
214 N=18 :REM LAENGE DES RECORD                <172>
220 PC=55620:REM FARBRAM                       <176>
230 XY=PA+X+Y*40 :REM SCREEN-POS               <209>
240 CO=PC+X+Y*40 :REM COLOR                    <151>
244 FOR I=0 TO 7:READ F(I):NEXT                <100>
246 DATA 128,64,32,16,8,4,2,1                  <006>
248 DIM P(11):REM OUT FOR PRINTER              <113>
260 AS=32               :REM ASCI-CODE         <213>
262 AB=32               :REM ASCI-ANFANG       <188>
264 AA=1                :REM ASCI-ANFANG       <167>
266 AE=254              :REM ASCI-ENDE         <064>
270 LS=12               :REM LINE-SPACE        <100>
272 SA=0                :REM SCHRIFTART        <134>
280 RO$=CHR$(5)+"ROM"+CHR$(144)+"/RAM"         <227>
282 RA$="ROM/"+CHR$(5)+"RAM"+CHR$(144)         <202>
284 RM$=RO$:FL=1                               <008>
```

Listing. Zeichensatzgenerator »MAKE-CHR«

```
999 :                                       <213>
1000 REM MENUE *************************    <161>
1010 PRINT"{CLR}":GX$=""                    <186>
1015 PRINT CHR$(31);"{SPACE,RVSON}£U*****I
     {22SPACE}U*****I¥{RVOFF}";CHR$(144)    <242>
1020 PRINT CHR$(31);"{SPACE,RVSON,SPACE}*{
     2SPACE}*{2SPACE}Ū***** MAKE-CHR{2SPAC
     E}C*****Ŵ{2SPACE}*{2SPACE}*{SPACE,RVO
     FF}";CHR$(144)                         <248>
1025 PRINT CHR$(31);" ¥{RVSON,3SPACE}J**K{
     22SPACE}J**K{3SPACE,RVOFF}£";CHR$(144
     )                                      <239>
1030 PRINT:PRINT                            <222>
1040 PRINT TAB(5);"{DOWN}(1) - ERSTELLE ZE
     ICHENSATZ                              <021>
1050 PRINT TAB(5);"{DOWN}(2) - SENDE ZEICH
     ENSATZ ZU DRUCKER"                     <190>
1060 PRINT TAB(5);"{DOWN}(3) - DRUCKER-HAN
     DLING "                                <017>
1070 PRINT TAB(5);"{DOWN}(4) - ";CHR$(155)
     ;"OEFFNEN EINER REL-DATEI";CHR$(144)   <115>
1084 PRINT TAB(5);"{DOWN}(5) - ";CHR$(155)
     ;"COPY SINGLE CHR ";CHR$(144)          <140>
1086 PRINT TAB(5);"{DOWN}(6) - ";CHR$(155)
     ;"COPY CHR-DATEI ";CHR$(144)           <195>
1090 PRINT TAB(5);"{3DOWN}JOYSTICK IN PORT
      2 !"                                  <135>
1100 GET G$                                 <086>
1101 GZ=GZ+G                                <137>
1102 IF GZ=420 THEN G=-1:PRINT"{HOME}"SPC(
     14)"{2DOWN}BY R.BUESCH"                <201>
1104 IF GZ= 0 THEN G= 1:PRINT"{HOME}"SPC(1
     4)"{2DOWN,SPACE}MAKE-CHR{2SPACE}"      <019>
1108 IF G$="" GOTO 1100                     <040>
1110 IF G$="1" THEN 7000                    <085>
1120 IF G$="2" THEN 6000                    <220>
1130 IF G$="3" THEN 2000                    <086>
1140 IF G$="4" THEN 5000                    <237>
1160 IF G$="5" THEN 10500                   <205>
1170 IF G$="6" THEN 12000                   <024>
1190 GOTO 1100                              <130>
1999 :                                      <197>
2000 REM DRUCKER-HANDLING *************     <075>
2020 PRINT"{CLR,DOWN,SPACE,RVSON,SPACE}***
     **** DRUCKER-HANDLING ***********{SPA
     CE,RVOFF}"                             <051>
2030 PRINT:PRINT                            <204>
2040 PRINT TAB(6);"{DOWN}(1) - COPY CHR RO
     M -> RAM"                              <143>
2042 PRINT TAB(6);"{DOWN}(2) - WAEHLE ";RM
     $                                      <122>
2050 PRINT TAB(6);"{DOWN}(3) - WAEHLE LINE
     -SPACE"                                <101>
2054 PRINT TAB(6);"{DOWN}(4) - WAEHLE SCHR
     IFTART"                                <058>
2060 PRINT TAB(6);"{DOWN}(5) - ";CHR$(155)
     ;"TESTE DRUCKER ";CHR$(144)            <163>
2100 GET G$:IF G$=""THEN 2100               <136>
2110 IF G$="1" THEN GOSUB 9100:GOTO 2020    <097>
2120 IF G$="2" THEN GOSUB 9120:GOTO 2020    <044>
2132 IF G$="3" THEN GOSUB 9160:GOTO 2020    <056>
2134 IF G$="4" THEN GOSUB 9180:GOTO 2020    <252>
2136 IF G$="5" THEN GOSUB 6500:GOTO 2020    <165>
2140 IF G$="←" THEN 1000 :REM MENUE         <197>
2160 GOTO 2000                              <076>
4999 :                                      <149>
5000 REM OEFFNE RECORD DATEI **********     <239>
5010 PRINT"{CLR}"                           <172>
5020 PRINT"{2SPACE}NAME DER RECORD DATEI ?
     "                                      <107>
5030 PRINT"{2SPACE,RVSON,SPACE}";NA$;"{SPA
     CE,RVOFF,SPACE}";CHR$(5);              <080>
5040 GX$=NA$:GOSUB 15000:REM INPUT          <008>
5041 IF GX$="←" GOTO 1000                   <108>
5042 NA$=GX$:PRINT CHR$(144)                <228>
5044 NX$=NA$                                <196>
5050 PRINT:PRINT CHR$(31);"{2SPACE,RVSON,S
     PACE}";NA$;"{SPACE,RVOFF,SPACE}";CHR$
     (144);"WIRD ERSTELLT"                  <164>
5060 GOSUB 10000                            <214>
5064 OPEN 2,8,15                            <158>
5070 GOSUB 6700:REM DISC-ERROR              <223>
5085 CLOSE 2                                <022>
5090 GOTO 1000                              <204>
5999 :                                      <133>
6000 REM SEND RECORD TO PRINTER *******     <069>
6010 PRINT"{CLR,2RIGHT,DOWN}WELCHER ZEICHE
     NSATZ SOLL UEBERTRAGEN                 <114>
6012 PRINT"{2SPACE}WERDEN{2SPACE}?          <209>
6013 PRINT"{DOWN,2SPACE,RVSON,SPACE}";NA$;
     "{SPACE,RVOFF,SPACE}";CHR$(5);         <083>
6020 GX$=NA$:GOSUB 15000:REM INPUT          <228>
6021 IF GX$="←" GOTO 1000                   <072>
6022 NA$=GX$:PRINT CHR$(144)                <192>
6023 NX$=NA$                                <159>
6024 IF LEFT$(NA$,3)="BIG"THEN LS=8         <111>
6025 PRINT"{DOWN,3SPACE}WELCHE ASCI-CODES
     ?":PRINT                               <141>
6026 PRINT"{2SPACE}VON{2SPACE,RVSON}";AA;"
     {LEFT,SPACE,RVOFF,SPACE}";CHR$(5);     <165>
6027 GA=1:GX$=STR$(AA):GOSUB 15000:REM INP
     UT                                     <047>
6028 IF GX$="←"GOTO 1000                    <079>
6029 AA=VAL(GX$)                            <158>
6030 IF AA>256 GOTO 6027                    <015>
6032 PRINT CHR$(144):PRINT                  <052>
6040 PRINT"{2SPACE}BIS{2SPACE,RVSON}";AE;"
     {LEFT,SPACE,RVOFF,SPACE}";CHR$(5);     <019>
6042 GA=1:GX$=STR$(AE):GOSUB 15000:REM INP
     UT                                     <063>
6044 IF GX$="←"GOTO 1000                    <095>
6046 AE=VAL(GX$)                            <207>
6048 IF AE>256 GOTO 6042                    <225>
6050 PRINT CHR$(144):PRINT                  <070>
6054 OPEN 1,8,2,NA$                         <182>
6056 OPEN 2,8,15                            <134>
6058 GOSUB 6700:REM DISC-ERROR              <195>
6060 AS=AA                                  <028>
6070 PRINT#2,"P"+CHR$(98)+CHR$(AS)+CHR$(0)
     +CHR$(1)                               <243>
6075 FOR I=1 TO 17:IF VA=1 THEN GOTO 6180   <220>
6080 GET#1,RC$:IF RC$=""THEN RC$=CHR$(0)    <026>
6120 IF I=1 THEN GOTO 6300:REM TESTE ZEICH
     EN                                     <232>
6150 PRINT#3,RC$;                           <090>
6170 RC$=""                                 <222>
6180 NEXT I                                 <168>
6182 GOSUB 6700:REM DISC-ERROR              <065>
6190 PRINT CHR$(5);"{2SPACE}ASCI ={SPACE,R
     VSON}";AS;"{LEFT,SPACE,RVOFF,18SPACE,
     UP}";CHR$(144)                         <066>
6195 RK$="":VA=0                            <186>
6198 AS=AS+1                                <178>
6200 IF AS<=AE GOTO 6070                    <193>
6220 CLOSE 2:CLOSE 1                        <146>
6230 GOSUB 9140:REM SEL.RAM                 <021>
6240 GOSUB 6500:REM TESTE PRINTER           <142>
6298 GOTO 1000:REM MENUE                    <088>
6300 REM TESTE ZEICHEN ****************     <253>
6310 IF RC$=CHR$(255)THEN PRINT SPC(17);"R
     ECORD EMPTY{UP}":RC$="":VA=1:GOTO 618
     0                                      <197>
6320 GOTO 6150                              <124>
6499 :                                      <125>
6500 REM TEST DRUCKER *****************     <240>
6505 GOSUB 9194:REM SET SCHRIFT             <232>
6506 GOSUB 9174:REM SET LINE-SPACE          <198>
6510 FOR I=1 TO 64 ;GOSUB 6540:NEXT:PRINT#
     3,CHR$(10)                             <229>
6512 FOR I=128 TO 192:GOSUB 6540:NEXT:PRIN
     T#3,CHR$(10)                           <247>
6514 FOR I= 65 TO 96:GOSUB 6540:NEXT:PRINT
     #3,CHR$(10)                            <136>
6516 FOR I=193 TO 224:GOSUB 6540:NEXT:PRIN
     T#3,CHR$(10)                           <048>
6518 FOR I= 97 TO 126:GOSUB 6540:NEXT:PRIN
     T#3,CHR$(10)                           <155>
6520 FOR I=225 TO 254:GOSUB 6540:NEXT:PRIN
     T#3,CHR$(10)                           <159>
6530 PRINT#3,CHR$(13)                       <144>
6532 RETURN                                 <240>
6540 IF I=7  THEN I=28                      <202>
6542 IF I=127 THEN I=128                    <102>
6544 IF I=135 THEN I=156                    <150>
6580 PRINT#3,CHR$(I);                       <232>
6590 RETURN                                 <042>
6600 REM BEEP *************************     <102>
6620 FOR I=1 TO 3                           <113>
6630 PRINT#3,CHR$(7);:FOR K=1 TO 150:NEXT   <001>
6640 NEXT I                                 <118>
6660 RETURN                                 <114>
6699 :                                      <071>
6700 REM INPUT DISC-ERROR *************     <035>
6710 INPUT#2,E,E$,E1,E2                     <052>
6720 IF E<>0 THEN CLOSE 1:CLOSE 2:GOTO 680
     0                                      <074>
```

Listing. Zeichensatzgenerator (Fortsetzung)

```
6730 RETURN                                          <184>
6800 PRINT"{CLR,2DOWN,2SPACE}ERROR ON DISC
     : ";CHR$(5);E$;CHR$(144)                        <054>
6802 PRINT"{2DOWN,2SPACE}PRESS ANY KEY TO
     RESTART !                                       <051>
6810 GET G$:IF G$=""GOTO 6810                        <242>
6820 RUN                                             <002>
6999 :                                               <117>
7000 REM ERSTELLE ZEICHEN *************              <192>
7010 PRINT"{CLR}"                                    <140>
7020 PRINT"{2SPACE}AUS WELCHER DATEI WILLS
     T DU LESEN ?"                                   <179>
7022 PRINT"{2SPACE,RVSON,SPACE}";NX$;"{SPA
     CE,RVOFF,SPACE}";CHR$(5);                       <063>
7030 GX$=NX$:GOSUB 15000:REM INPUT                   <106>
7031 IF GX$="←" GOTO 1000                            <066>
7032 NX$=GX$:PRINT CHR$(144):PRINT:PRINT             <182>
7050 IF NY$=""THEN NY$=NX$                           <037>
7052 PRINT"{2SPACE}IN WELCHE DATEI WILLST
     DU SCHREIBEN ?"                                 <168>
7054 PRINT"{2SPACE,RVSON,SPACE}";NY$;"{SPA
     CE,RVOFF,SPACE}";CHR$(5);                       <096>
7058 GX$=NY$:GOSUB 15000:REM INPUT                   <006>
7059 IF GX$="←" GOTO 1000                            <094>
7060 NY$=GX$:PRINT CHR$(144):PRINT                   <112>
7070 PRINT"{CLR}"                                    <200>
7100 GOSUB 10100:REM ZEICHENEINGABE                  <215>
7110 IF GX$="←" GOTO 1000:REM MENU                   <179>
7199 :                                               <063>
7200 REM ABFRAGE **********************              <240>
7210 G$="":GET G$                                    <115>
7215 IF G$="N" THEN GOSUB 9900:GOSUB 8700
     :REM LOESCHE & ANFANGSPOS                       <199>
7218 IF G$="9" THEN GOSUB 8500
     :REM 9-DOTS                                     <161>
7220 IF G$="←" GOTO 1000:REM MENUE                   <137>
7222 IF G$="F" THEN GOSUB 11000:GOSUB 8700
     :REM FILL & ANFANGSPOS                          <211>
7230 IF G$=CHR$(13) THEN GOSUB 10100:REM E
     INGABE                                          <022>
7348 IF G$=CHR$(136)THEN GOSUB 9400:GOTO 8
     800:REM LESE NACHSTES ZEICHEN                   <040>
7350 IF G$=CHR$(133)THEN GOTO 7400:REM SAV
     EN                                              <007>
7360 IF G$="{UP}" THEN DY=-1:GOTO 8400               <234>
7362 IF G$="{DOWN}" THEN DY= 1:GOTO 8400             <218>
7364 IF G$="{LEFT}" THEN DX=-1:GOTO 8400             <236>
7366 IF G$="{RIGHT}" THEN DX= 1:GOTO 8400            <220>
7367 IF G$="*" THEN GOSUB 9800                       <236>
7368 IF G$="*" THEN GOSUB 9800                       <162>
7390 GOTO 7500: REM JOYSTICK                         <158>
7399 :                                               <007>
7400 GOSUB 9000 :REM ALTES FELD                      <250>
7410 GOSUB 8600 :REM WRITE RECORD                    <242>
7415 A=139:GOSUB 8550:REM RESET 9DOTS                <138>
7420 AS=AS+1:REM INCREM. ASCI                        <163>
7450 GOTO 7100 :REM NEUE EINGABE                     <208>
7499 :                                               <109>
7500 REM JOYSTICK *********************              <000>
7600 P =PEEK(56320)                                  <249>
7610 JU= P AND 1                                     <127>
7620 JD= (P AND 2)/2                                 <047>
7630 JL= (P AND 4)/4                                 <139>
7640 JR= (P AND 8)/8                                 <233>
7650 FR= (P AND 16)/16                               <105>
7680 FOR I=1 TO 300:NEXT                             <043>
7690 :                                               <046>
7720 IF JU=0   THEN DY=-1:GOTO 8400                  <183>
7730 IF JD=0   THEN DY= 1:GOTO 8400                  <060>
7740 IF JL=0   THEN DX=-1:GOTO 8400                  <056>
7750 IF JR=0   THEN DX= 1:GOTO 8400                  <047>
7755 IF FR=1 THEN FA=1                               <156>
7760 IF FR=0 THEN GOSUB 8200:REM TEST-FR             <054>
7790 GOTO 7210 :REM ABFRAGE                          <007>
7799 :                                               <155>
8200 REM TEST-FR **********************              <217>
8210 IF FR<>FA THEN FA=0:GOSUB 9800:RETURN
     :REM STAR/BLANK                                 <119>
8220 IF M=0 THEN GOSUB 9600:RETURN:REM STA
     R                                               <244>
8230 GOSUB 9700:RETURN:REM BLANK                     <047>
8299 :                                               <223>
8400 REM PRINT*************************              <064>
8410 IF Y+DY>7 THEN DY=0: GOTO 7210                  <229>
8415 IF Y+DY<0 THEN DY=0: GOTO 7210                  <231>
8420 IF X+DX>10 THEN DX=0: GOTO 7210                 <185>
8425 IF X+DX<0 THEN DX=0: GOTO 7210                  <201>
8430 GOSUB 9400                                      <151>
8432 X=X+DX:Y=Y+DY                                   <120>
8440 XY=PA+X+Y*40                                    <087>
8442 CO=PC+X+Y*40                                    <071>
8450 GOSUB 9500                                      <168>
8455 DX=0:DY=0                                       <123>
8460 GOTO 7210                                       <130>
8499 :                                               <093>
8500 REM ATTRIBUTE A ******************              <009>
8504 IF A=139 THEN A=11:GOSUB 8550:RETURN            <216>
8508 IF A=11 THEN A=139:GOSUB 8550:RETURN            <154>
8510 RETURN                                          <186>
8550 IF A=139 THEN DO$="{18SPACE}"                   <029>
8560 IF A=11  THEN DO$=CHR$(5)+"{3SPACE,RV
     SON,SPACE}9DOT ACTIV{2SPACE,RVOFF}"+C
     HR$(144)                                        <237>
8570 PRINT"{HOME,5DOWN}";DO$                         <028>
8590 RETURN                                          <010>
8599 :                                               <193>
8600 REM WRITE RECORD *****************              <107>
8604 NA$=NY$                                         <028>
8610 XR$=CHR$(27)+"&"+CHR$(0)+CHR$(AS)+CHR
     $(AS)+CHR$(A)                                   <080>
8640 FOR I=0 TO 10:YR$=YR$+CHR$(P(I)):NEXT           <102>
8645 ZR$=XR$+YR$                                     <139>
8660 OPEN 1,8,2,NA$                                  <248>
8662 OPEN 2,8,15                                     <200>
8664 PRINT#2,"P"+CHR$(98)+CHR$(AS)+CHR$(0)
     +CHR$(1)                                        <041>
8666 PRINT#1,ZR$                                     <186>
8670 GOSUB 6700:REM DISC-ERROR                       <011>
8680 CLOSE 1:CLOSE 2                                 <120>
8682 FOR I=0 TO 10:P(I)=0:NEXT                       <080>
8684 YR$=""                                          <086>
8690 RETURN                                          <110>
8700 REM ANFANGSPOSITION **************              <207>
8710 X=0:Y=0                                         <105>
8712 XY=PA+X+Y*40                                    <017>
8714 CO=PC+X+Y*40                                    <239>
8750 GOSUB 9500:REM NEUES FELD                       <162>
8760 RETURN                                          <182>
8800 REM READ & DISPLAY ***************              <107>
8802 NA$=NX$                                         <100>
8804 GOSUB 9900: REM NEW                             <099>
8810 OPEN 1,8,2,NA$                                  <144>
8812 OPEN 2,8,15                                     <096>
8820 PRINT#2,"P"+CHR$(98)+CHR$(AS)+CHR$(0)
     +CHR$(6)                                        <023>
8830 FOR I= 0 TO 11                                  <053>
8835 GOSUB 6700:REM DISC-ERROR                       <178>
8840 GET#1,ZA$                                       <093>
8841 ZA(I)=ASC(ZA$+CHR$(0))                          <217>
8842 IF ZA(0)=139 THEN A=139:GOSUB 8550              <157>
8843 IF ZA(0)=11 THEN A=11 :GOSUB 8550               <034>
8844 NEXT                                            <218>
8846 CLOSE 1:CLOSE 2                                 <032>
8847 :                                               <187>
8850 FOR X=0 TO 10                                   <192>
8852 FOR Y=0 TO 7                                    <157>
8854 XY=PA+X+Y*40                                    <159>
8856 CO=PC+X+Y*40                                    <125>
8860 IF F(Y)=(ZA(X+1)AND F(Y)) THEN POKE X
     Y,42:POKE CO,1                                  <050>
8864 NEXT Y                                          <184>
8868 NEXT X                                          <180>
8885 GOSUB 8700:REM ANFANGSPOS                       <160>
8890 GOTO 7100:REM NEUEINGABE                        <150>
8899 :                                               <239>
9000 REM UMRECHNUNG *******************              <099>
9004 GOSUB 9400: REM ALTES FELD                      <140>
9010 FOR X=0 TO 10                                   <098>
9020 FOR Y=0 TO 7                                    <071>
9022 XY=PA+X+Y*40                                    <073>
9030 IF PEEK(XY)=42 THEN P(X)=P(X)+F(Y)              <005>
9040 NEXT Y:NEXT X                                   <115>
9090 RETURN                                          <002>
9100 REM COPY CHR-ROM TO RAM **********              <163>
9104 PRINT#3,CHR$(27);":";CHR$(0);CHR$(0);
     CHR$(0);CHR$(7);:RETURN                         <009>
9120 REM SELECT CHR-ROM/RAM ***********              <095>
9124 IF FL=1 GOTO 9140                               <096>
9126 RM$=RO$:FL=1                                    <214>
9130 PRINT#3,CHR$(27);"%";CHR$(0);CHR$(0);
     CHR$(7):RETURN                                  <090>
9140 RM$=RA$:FL=0:REM SEL.RAM                        <049>
9144 PRINT#3,CHR$(27);"%";CHR$(1);CHR$(0);
     CHR$(7);:RETURN                                 <188>
```

Listing. Zeichensatzgenerator (Fortsetzung). Bitte bei der Eingabe den Checksummer auf Seite 6 beachten.

```
9160 REM LINE-SPACE ******************          <054>
9164 PRINT"{CLR,DOWN,2SPACE}WELCHEN ZEILEN
     ABSTAND WILLST DU ?                         <170>
9168 PRINT"{DOWN,2SPACE}(0-85){SPACE,RVSON
     }"LS"{LEFT,SPACE,RVOFF,SPACE}"CHR$(5)
     ;                                           <066>
9170 GX$=STR$(LS):GOSUB 15000:LS=VAL(GX$):
     IF LS>85 THEN 9160                          <184>
9172 PRINT CHR$(144):PRINT                       <144>
9174 PRINT#3,CHR$(27);"A";CHR$(LS);CHR$(7)
     ;                                           <200>
9178 RETURN                                      <090>
9180 REM SCHRIFTART ******************          <023>
9182 PRINT"{CLR,DOWN,2SPACE}WELCHE SCHRIFT
     ART WILLST DU ?                             <074>
9184 PRINT"{DOWN,2SPACE}(0-64){SPACE,RVSON
     }"SA"{LEFT,SPACE,RVOFF,SPACE}"CHR$(5)
     ;                                           <171>
9186 GX$=STR$(SA):GOSUB 15000:SA=VAL(GX$):
     IF SA>64 THEN 9180                          <243>
9188 PRINT CHR$(144):PRINT                       <160>
9194 PRINT#3,CHR$(27);"!";CHR$(SA);CHR$(7)
     ;                                           <160>
9198 RETURN                                      <110>
9398 END                                         <000>
9399 :                                           <231>
9400 REM ALTES FELD ******************          <030>
9410 IF PEEK(XY)=170 THEN POKE XY,42:POKE
     CO,1                                        <140>
9420 IF PEEK(XY)=160 THEN POKE XY,32:POKE
     CO,1                                        <144>
9450 RETURN                                      <108>
9499 :                                           <077>
9500 REM NEUES FELD ******************          <156>
9510 IF PEEK(XY)=42  THEN POKE XY,170:POKE
      CO,1                                       <101>
9520 IF PEEK(XY)=32  THEN POKE XY,160:POKE
      CO,1                                       <107>
9550 RETURN                                      <210>
9599 :                                           <177>
9600 REM PRINT STAR ******************          <146>
9610 POKE XY,170:POKE PC,1                       <126>
9650 RETURN                                      <054>
9699 :                                           <021>
9700 REM PRINT BLANK *****************          <199>
9710 POKE XY,160:POKE PC,1                       <098>
9750 RETURN                                      <156>
9799 :                                           <123>
9800 REM PRINT BLANK/STAR ************          <147>
9810 IF PEEK(XY)=170 THEN POKE XY,160:POKE
      CO,1:M=1:RETURN                            <241>
9820 IF PEEK(XY)=160 THEN POKE XY,170:POKE
      CO,1:M=0:RETURN                            <219>
9899 :                                           <223>
9900 REM NEW *************************          <124>
9910 FOR Y=0 TO 7                                <199>
9915 FOR X=0 TO 10                               <241>
9916 XY=PA+X+Y*40                                <205>
9918 CO=PC+X+Y*40                                <171>
9920 POKE XY,32:POKE CO,5                        <131>
9930 NEXT X:NEXT Y                               <043>
9950 RETURN                                      <100>
9999 :                                           <069>
10000 REM OEFFNE RECORD DATEI *********          <090>
10010 OPEN 1,8,2,NA$+",L,"+CHR$(18)              <011>
10020 OPEN 2,8,15                                <034>
10030 PRINT#2,"P"+CHR$(98)+CHR$(255)+CHR$(
      0)+CHR$(1)                                 <194>
10040 PRINT#1,CHR$(255):REM FREIGEBEN            <115>
10050 PRINT#2,"I"                                <236>
10060 CLOSE 1:CLOSE 2                            <232>
10070 RETURN                                     <222>
10099 :                                          <169>

10100 REM ZEICHENEINGABE **************          <192>
10104 GOSUB 9400 :REM ALTES FELD                 <224>
10110 PRINT"{HOME,SPACE,RVSON,SPACE}";NX$;
      "->";NY$;"{SPACE,RVOFF}";TAB(25);"{R
      VSON,SPACE}ASCI{SPACE,RVOFF}";TAB(33
      );"{RVSON,SPACE}CHR{SPACE,RVOFF}"          <230>
10111 PRINT CHR$(5);TAB(25);AS;"{LEFT,3SPA
      CE}";TAB(35);                              <106>
10112 IF AS>32 AND AS<128 THEN PRINT CHR$(
      AS);:GOTO 10120                            <031>
10114 IF AS>160 AND AS<256 THEN PRINT CHR$
      (AS);:GOTO 10120                           <135>
10116 PRINT TAB(34);"{2SPACE}";                  <071>
10120 PRINT:PRINT CHR$(144)                      <128>
10130 PRINT"{2SPACE}WHICH ASCI-CODE{9SPACE
      ,8LEFT}";CHR$(5);                          <112>
10140 GX$=STR$(AS):GA=1:GOSUB 15000:REM IN
      PUT                                        <145>
10142 IF GX$="←" THEN RETURN                     <207>
10144 AS=VAL(GX$):PRINT CHR$(144)                <165>
10146 IF AS=0 THEN 10100                         <107>
10150 IF AS>255 GOTO 10100                       <111>
10160 PRINT"{DOWN}";DO$;"{DOWN}"                 <163>
10180 PRINT"{3SPACE}UCRCRCRCRCRCI{3SPACE,R
      VSON}£{16SPACE}¥{RVOFF}"                   <058>
10190 PRINT"{3SPACE}=";SPC(11);"={3SPACE,R
      VSON,2SPACE}** MAKE-CHR **{2SPACE,RV
      OFF}"                                      <111>
10200 PRINT"{3SPACE}=";SPC(11);"={3SPACE}¥
      {RVSON,16SPACE,RVOFF}£"                    <057>
10220 PRINT"{3SPACE}=";SPC(11);"=                <224>
10230 PRINT"{3SPACE}=";SPC(11);"={6SPACE}R
      ET = ASCI                                  <234>
10240 PRINT"{3SPACE}=";SPC(11);"={8SPACE}←
       = MENUE                                   <121>
10250 PRINT"{3SPACE}=";SPC(11);"={8SPACE}F
       = FILL                                    <154>
10260 PRINT"{3SPACE}=";SPC(11);"={8SPACE}N
       = NEW                                     <112>
10270 PRINT"{3SPACE}=";SPC(11);"={8SPACE}9
       = 9DOTS                                   <213>
10280 PRINT"{3SPACE}JCFCFCFCFCFCK";CHR$(5)
      ;"{7SPACE}F7";CHR$(144);" = READ           <109>
10290 PRINT TAB(23);CHR$(5);"F1";CHR$(144)
      ;" = SAVE                                  <055>
10292 PRINT"{3DOWN,2SPACE}PRINT YOUR CHARA
      CTER                                       <068>
10294 PRINT"{2SPACE}WITH";CHR$(5);" JOYSTI
      C 2";CHR$(144)                             <147>
10296 PRINT"{2SPACE}OR{2SPACE}";CHR$(5);"
      CRSR/*";CHR$(144)                          <202>
10310 GOSUB 8700:REM ANFANGSPOSITION             <239>
10320 RETURN                                     <218>
10399 :                                          <215>
10500 REM COPY FILES ******************          <150>
10550 PRINT"{CLR,RIGHT,DOWN,SPACE}AUS WELC
      HER DATEI WIRD COPIERT ?                   <115>
10560 PRINT"{2SPACE,RVSON,SPACE}";NX$;"{SP
      ACE,RVOFF,SPACE}";CHR$(5);                 <045>
10570 GX$=NX$:GOSUB 15000:REM INPUT              <090>
10571 IF GX$="←" GOTO 1000                       <050>
10572 NX$=GX$:PRINT CHR$(144):PRINT              <060>
10574 IF NX$="←"THEN 1000:REM MENUE              <175>
10580 PRINT"{2SPACE}VON ASCI{SPACE,RVSON}"
      AA"{LEFT,SPACE,RVOFF,SPACE}";CHR$(5)
      ;                                          <151>
10590 GA=1:GX$=STR$(AA):GOSUB 15000:REM IN
      PUT                                        <038>
10592 IF GX$="←"GOTO 1000                        <071>
10594 AA=VAL(GX$)                                <151>
10596 IF AA>256 GOTO 10590                       <114>
10598 PRINT CHR$(144):PRINT                      <046>
10600 PRINT"{2SPACE}BIS ASCI{SPACE,RVSON}"
      AE"{LEFT,SPACE,RVOFF,SPACE}";CHR$(5)
      ;                                          <236>
10605 GA=1:GX$=STR$(AE):GOSUB 15000:REM IN
      PUT                                        <054>
10606 IF GX$="←"GOTO 1000                        <085>
10607 AE=VAL(GX$)                                <196>
10608 IF AE>256 GOTO 10605                       <238>
10610 PRINT CHR$(144):PRINT"{3DOWN}"             <211>
10620 PRINT"{2SPACE}IN WELCHE DATEI WIRD C
      OPIERT ?"                                  <163>
10630 PRINT"{2SPACE,RVSON,SPACE}";NY$;"{SP
      ACE,RVOFF,SPACE}";CHR$(5);                 <116>
10640 GX$=NY$:GOSUB 15000:REM INPUT              <032>
10641 IF GX$="←" GOTO 1000                       <120>
10642 NY$=GX$:PRINT CHR$(144):PRINT              <138>
10644 IF NY$="←"THEN 1000:REM MENUE              <005>
10650 PRINT"{2SPACE}AB{2SPACE}ASCI{SPACE,R
      VSON}"AB"{LEFT,SPACE,RVOFF,SPACE}";C
      HR$(5);                                    <167>
10660 GA=1:GX$=STR$(AB):GOSUB 15000:REM IN
      PUT                                        <172>
10661 IF GX$="←"GOTO 1000                        <140>
10662 AB=VAL(GX$)                                <227>
10666 IF AB>256 GOTO 10590                       <200>
10668 PRINT CHR$(144):PRINT                      <116>
10670 PRINT CHR$(5);"{DOWN,7SPACE}**** I'A
      M BUSY !! ****{2DOWN}";CHR$(144)           <053>
10699 :                                          <005>
10700 REM COPY ************************          <193>
10730 YR$="":XR$=""                              <175>
```

Listing. Zeichensatzgenerator (Fortsetzung)

```
10740 OPEN 2,8,15                                  <246>
10750 FOR I=AA TO AE                               <096>
10760 OPEN 1,8,2,NX$                               <154>
10762 GOSUB 6700:REM DISC-ERROR                    <073>
10780 PRINT#2,"P"+CHR$(98)+CHR$(I)+CHR$(0)
      +CHR$(6)                                     <085>
10784 FOR K=0 TO 11                                <106>
10790 GET#1,G$:IF G$=""THEN G$=CHR$(0)             <086>
10794 YR$=YR$+G$                                   <067>
10798 NEXT K                                       <230>
10800 GOSUB 6700:REM DISC-ERROR                    <111>
10806 CLOSE 1                                      <149>
10808 :                                            <116>
10810 OPEN 1,8,2,NY$                               <208>
10812 GOSUB 6700:REM DISC-ERROR                    <123>
10814 AS=AB+I-AA                                   <024>
10816 XR$=CHR$(27)+"&"+CHR$(0)+CHR$(AS)+CH
      R$(AS)                                       <139>
10817 ZR$=XR$+YR$                                  <025>
10818 PRINT"{9SPACE}ASCI ={SPACE,RVSON}"AS
      "{LEFT,SPACE}←"I"{LEFT,SPACE,RVOFF,U
      P,2SPACE}"                                   <140>
10820 PRINT#2,"P"+CHR$(98)+CHR$(AS)+CHR$(0
      )+CHR$(1)                                    <167>
10830 PRINT#1,ZR$                                  <064>
10832 GOSUB 6700:REM DISC-ERROR                    <143>
10840 CLOSE 1                                      <183>
10850 XR$="":YR$="":ZR$=""                         <205>
10890 NEXT I                                       <050>
10894 CLOSE 2                                      <245>
10910 GOTO 1000:REM MENUE                          <128>
10999 :                                            <051>
11000 REM FILL ************************            <020>
11010 FOR Y=0 TO 7                                 <029>
11020 FOR X=0 TO 10 STEP 2                         <103>
11030 XY=PA+X+Y*40                                 <049>
11040 CO=PC+X+Y*40                                 <120>
11044 POKE XY,42:POKE CO,1                         <178>
11050 NEXT X:NEXT Y                                <149>
11060 RETURN                                       <196>
12000 REM COPY-DATEI *******************           <028>
12050 PRINT"{CLR,DOWN,2SPACE}DU KANNST JET
      ZT CHR-DATEIEN AUF EINE"                     <116>
12060 PRINT"{2SPACE}ANDERE DISKETTE COPIER
      EN !                                         <067>
12070 PRINT"{2DOWN,2SPACE}WELCHE DATEI WIR
      D COPIERT ?"                                 <047>
12080 PRINT"{2SPACE,RVSON,SPACE}"NX$;"{SPA
      CE,RVOFF,SPACE}";CHR$(5);                    <224>
12090 GX$=NX$:GOSUB 15000:REM INPUT                <086>
12091 IF GX$="←" GOTO 1000                         <046>
12092 NX$=GX$:PRINT CHR$(144):PRINT:PRINT          <162>
12094 IF NY$=""THEN NY$=NX$                        <001>
12100 PRINT"{2SPACE}NEUER DATEINAME ?              <160>
12110 PRINT"{2SPACE,RVSON,SPACE}"NY$;"{SPA
      CE,RVOFF,SPACE}";CHR$(5);                    <126>
12120 GX$=NY$:GOSUB 15000:REM INPUT                <244>
12121 IF GX$="←" GOTO 1000                         <076>
12122 NY$=GX$:PRINT CHR$(144):PRINT                <094>
12130 PRINT"{2DOWN,10SPACE}READING{SPACE,R
      VSON}";CHR$(31);NX$;CHR$(5)                  <113>
12140 PRINT"{DOWN,7SPACE}**** COFFE-BREAK
      ****{DOWN}";CHR$(144)                        <005>
12199 :                                            <237>
12210 OPEN 1,8,2,NX$                               <078>
12220 OPEN 2,8,15                                  <202>
12222 GOSUB 6700:REM DISC-ERROR                    <007>
12224 FOR K=1 TO 255                               <176>
12228 PRINT SPC(12);"ASCI = ";K;"{LEFT,SPA
      CE,UP}"                                      <250>
12230 PRINT#2,"P"+CHR$(98)+CHR$(K)+CHR$(0)
      +CHR$(1)                                     <229>
12232 FOR I=0 TO 16                                <158>
12240 GET#1,G$:IF G$=""THEN G$=CHR$(0)             <010>
12244 IF I=0 AND G$=CHR$(255) THEN RC$=CHR
      $(255):I=16                                  <032>
12250 RC$(K)=RC$(K)+G$                             <229>
12260 NEXT I                                       <150>
12270 GOSUB 6700 :REM DISC-ERROR                   <055>
12280 NEXT K                                       <186>
12290 CLOSE 1:CLOSE 2                              <176>
12299 :                                            <083>
12300 REM WRITE                                    <105>
12310 GOSUB 6600:REM BEEP                          <070>
12350 PRINT"{CLR,DOWN,2SPACE}BITTE ANDERE
      DISKETTE EINLEGEN !{UP}"                     <248>
12360 TD$="DISKETTE{UP}"                           <173>
12370 PRINT SPC(15);TD$                            <089>
12372 FOR N=0 TO 400                               <048>
12380 GET G$:IF G$<>""THEN GOTO 12500              <217>
12382 NEXT                                         <200>
12400 PRINT SPC(15);CHR$(5);TD$;CHR$(144)          <165>
12410 FOR N=0 TO 400                               <086>
12420 GET G$:IF G$<>""THEN GOTO 12500              <001>
12430 NEXT                                         <248>
12440 GOTO 12370                                   <021>
12499 :                                            <027>
12500 REM COPY ON NEW DISC ************            <150>
12550 PRINT CHR$(144)                              <004>
12560 PRINT"{CLR,DOWN,10SPACE}WRITING{SPAC
      E,RVSON}";CHR$(31);" ";NY$;"{SPACE,R
      VOFF}"                                       <172>
12570 PRINT TAB(5);CHR$(5);"{2DOWN}**** IT
      'S TEE TIME NOW ****";CHR$(144);"{5D
      OWN}"                                        <235>
12580 PRINT TAB(10);"CREATING{SPACE,RVSON,
      SPACE}";NY$;"{SPACE,UP}"                     <137>
12582 NA$=NY$                                      <198>
12584 GOSUB 10000:REM OEFFNE RECORD DATEI          <091>
12599 :                                            <129>
12600 OPEN 1,8,2,NY$                               <220>
12604 OPEN 2,8,15                                  <078>
12608 GOSUB 6700:REM DISC-ERROR                    <141>
12610 FOR K=1 TO 255                               <054>
12612 PRINT TAB(10);"{3SPACE}ASCI =";K;"{L
      EFT,10SPACE,UP}"                             <038>
12620 PRINT#2,"P"+CHR$(98)+CHR$(K)+CHR$(0)
      +CHR$(1)                                     <111>
12650 PRINT#1,RC$(K)                               <007>
12654 GOSUB 6700:REM DISC-ERROR                    <187>
12660 NEXT K                                       <058>
12670 CLOSE 1:CLOSE 2                              <046>
12680 GOSUB 6600:REM BEEP                          <186>
12690 PRINT"{3DOWN,13SPACE}FERTIG !                <033>
12692 GET G$:IF G$=""THEN 12692                    <202>
12698 RUN                                          <038>
15000 REM GET-INPUT ******************             <161>
15010 G$=" ? ":GY$=GX$:GX$="":GZ$="{RVSON,
      SPACE,RVOFF}":G=1:PRINT" ";                  <136>
15020 PRINT"{LEFT}"G$;GZ$;                         <157>
15030 GET G$                                       <044>
15032 GZ=GZ+G                                      <096>
15033 IF GZ=0 THEN G=1:GZ$="{RVSON,SPACE,R
      VOFF}":PRINT CHR$(20);GZ$;                   <193>
15034 IF GZ=200 THEN G=-1:GZ$=CHR$(144)+"{
      RVSON,SPACE,RVOFF}"+CHR$(5):PRINT CH
      R$(20);GZ$;                                  <218>
15035 IF G$=""GOTO 15030                           <065>
15036 :                                            <024>
15038 IF GA=1 THEN GOTO 15200                      <124>
15040 IF G$=CHR$(13) GOTO 15180                    <017>
15042 IF G$="←" THEN GX$=G$:RETURN                 <143>
15050 IF G$=CHR$(20)  GOTO 15150                   <148>
15051 IF LEN(GX$)>9 GOTO 15030                     <103>
15053 IF G$=CHR$(34) GOTO 15030                    <095>
15060 IF G$="{UP}" GOTO 15030                      <175>
15070 IF G$="{DOWN}" GOTO 15030                    <121>
15080 IF G$="{CLR}" GOTO 15030                     <196>
15090 IF G$="{HOME}" GOTO 15030                    <142>
15100 IF G$=CHR$(141) GOTO 15030                   <209>
15120 IF G$=CHR$(148) GOTO 15030                   <003>
15130 GX$=GX$+G$                                   <187>
15140 GOTO 15020                                   <070>
15150 IF LEN(GX$)=0 THEN G$="":GOTO 15030          <068>
15160 GX$=LEFT$(GX$,LEN(GX$)-1)                    <208>
15170 GOTO 15020                                   <100>
15180 PRINT CHR$(20);                              <009>
15185 IF GX$="" THEN GX$=GY$                       <126>
15190 GA=0:RETURN                                  <249>
15200 REM EINGABE NUMERISCH ?                      <098>
15201 IF G$=CHR$(13)GOTO 15180                     <180>
15202 IF G$="←" THEN GX$=G$:GA=0:RETURN            <041>
15203 IF G$=CHR$(20)  GOTO 15150                   <047>
15209 IF LEN(GX$)>=3 GOTO 15030                    <098>
15210 IF G$="0" GOTO 15130                         <167>
15211 IF G$="1" GOTO 15130                         <040>
15212 IF G$="2" GOTO 15130                         <170>
15213 IF G$="3" GOTO 15130                         <043>
15214 IF G$="4" GOTO 15130                         <173>
15215 IF G$="5" GOTO 15130                         <046>
15216 IF G$="6" GOTO 15130                         <176>
15217 IF G$="7" GOTO 15130                         <049>
15218 IF G$="8" GOTO 15130                         <179>
15219 IF G$="9" GOTO 15130                         <052>
15230 GOTO 15030:REM GET                           <197>
```


Listing. Zeichensatzgenerator (Schluß)

Grafik 2000 — Eine Basic-Erweiterung, die es in sich hat

Diese Befehlserweiterung für den C 64 stellt Ihnen 41 neue Basic-Befehle und zwei neue Funktionen zur komfortablen Handhabung der mehrfarbigen HiRes-Grafik zur Verfügung.

Die Befehle sind, wie die Befehle des Basic V2, abkürzbar, vollkommen in den Interpreter integriert und werden selbstverständlich als Tokens gespeichert. Selbst nach »THEN« ist kein Doppelpunkt notwendig.

Da die zwei unabhängigen Grafikschirme und das Programm selbst außerhalb des Basic-Speichers liegen, stehen dem Benutzer weiterhin 38911 Bytes für seine Programme zur Verfügung. Auch Sprites benötigen keinen Basic-Speicherplatz.

Die Erweiterung selbst arbeitet außerordentlich schnell, da zum Beispiel das Programm zum Errechnen einer Punktadresse weniger als 1/15000 Sekunde benötigt. So dauert beispielsweise das AusFILLen des leeren Bildschirmes weniger als 50 Sekunden! Auch die weiteren Befehle arbeiten ähnlich schnell.

Diese Erweiterung ist zudem sehr sicher, denn selbst grobe Fehleingaben führen nicht zum Systemabsturz und somit Programmverlust (POKEs ausgenommen).

Grundlagen

Grafik 2000 unterstützt die farbige HiRes-Grafik des C 64. Diese Grafik ist aus 320 x 200 einzeln ansprechbaren Punkten aufgebaut. Dabei befindet sich der Punkt mit den Koordinaten 0/0 in der linken oberen und der Punkt mit den Koordinaten 319/199 in der rechten unteren Ecke des Bildschirmes.

Die Farbinformation für je 8 x 8 Punkte liegt im sogenannten Video-RAM. Hier steht in den oberen vier Bits die Farbe für einen gesetzten, in den unteren vier Bits die Farbe für einen gelöschten Punkt. Für den Anwender bedeutet dies, daß in einem 8 x 8 Punkte-Kästchen nur zwei verschiedene Farben vorkommen können.

Befehlsbesprechung

Anmerkung: Hinter dem Befehlsnamen steht die jeweilige Abkürzung, wobei »'« bedeutet, daß der folgende Buchstabe geSHIFTet eingegeben werden muß.

CLEAR — CL'E
Löscht den Grafikbildschirm. Die Farbe bleibt unverändert.

MODE n — M'O
Wenn n=1 ist, wird der Grafikmodus eingeschaltet; ist n=0, wird auf den Textbildschirm geschaltet. Dabei wird der Groß/Grafik-Modus eingeschaltet.

COLOR p,h — CO'L
Setzt die Farben im Video-RAM. Dabei steht p für die Punktfarbe und h für die Hintergrundfarbe. Es können Werte von 0 bis 15 für die Farben stehen.

CHANGE — CH'A
CHANGE wechselt zwischen den beiden Grafikschirmen, indem die jeweiligen Punkte ausgetauscht werden.

INVERS — IN'V
Invertiert die sichtbare Grafik. Das Video-RAM bleibt unverändert.

COMB n — CO'M
COMB führt eine logische Verknüpfung zwischen den beiden Grafikseiten aus. Das Ergebnis wird in die sichtbare Grafik geschrieben. Die Verknüpfungsart ist mit n wählbar:

n=1:	logisch OR
n=2:	logisch AND
n=3:	logisch EXOR

GSAVE a$,dv — G'S
GSAVE speichert die verdeckte Grafik mit dem Namen a$ an Gerät mit der Nummer dv ab. Die Parameter können entfallen.
Beispiel:

GASVE "BILD 1",8	Speichert die Grafik BILD 1 auf Diskette
GSAVE	Speichert die Grafik ohne Namen auf Kassette ab.

CSAVE a$,dv — C'S
Speichert das Video-RAM, die Farbe der Grafik, ab.
Die Parameter sind analog zu GSAVE.

GLOAD a$,dv — G'L
Mit GLOAD kann man die mit GSAVE abgespeicherte Grafik oder das mit CSAVE abgespeicherte Video-RAM laden.
Da dieser Befehl keine Basic-Zeiger verändert, eignet er sich auch zum Laden von Maschinenprogrammen, auch im Direktmodus.

SPOINT x,y — SP'O
Setzt einen Punkt an die Stelle x/y in der Grafik.

CPOINT x,y — C'P
Löscht den Punkt x/y.

IPOINT x,y — PS'C
Invertiert den Punkt x/y

PSCLINE x,y — PS'C
Setzt den Punkt x/y und löscht ab diesem Punkt eine Linie nach unten bis zum Grafikrand. Der Befehl ist besonders für 3D-Grafiken geeignet.

HMARK x,y — H'M
Setzt eine waagrechte »Markierung« an den Punkt x/y, das heißt, es wird eine 5 Punkte lange Linie gezeichnet, deren Mittelpunkt der Punkt x/y ist.

VMARK x,y — V'M
Analog HMARK, jedoch wird eine senkrechte Markierung gezeichnet.

HLINE y — H'L
Zeichnet eine durchgehende Horizontale an y.

VLINE x — V'L
Zeichnet eine durchgehende Vertikale an x.
Mit den letzten vier Befehlen ist es möglich, schnell und einfach Koordinatensysteme zu zeichnen.

SLINE x0,y0,x1,y1 — S'L
Zeichnet eine Linie von x0/y0 nach x1/y1. Die Richtung beziehungsweise Länge ist beliebig.

CLINE x0,y0,x1,y1 — CL'I
Wie SLINE, die Linie wird jedoch gelöscht.

ILINE x0,y0,x1,y1 — I'L
Wie SLINE, die Linie wird jedoch invertiert.

SCIRCLE x,y,a,b — S'C
Zeichnet eine Ellipse mit dem Mittelpunkt x/y und den Radien a und b.

CCIRCLE x,y,a,b C'C
Wie SCIRCLE, die Ellipse wird jedoch gelöscht.
ICIRCLE x,y,a,b I'C
Wie SCIRCLE, die Ellipse wird jedoch invertiert.
SELLIPSE x,y,a,b,dw,sw,ew,s S'E
Zeichnet einen Ellipsenbogen mit dem Mittelpunkt x/y und den Radien a und b.
Die weiteren Parameter bedeuten:

dw	Drehwinkel der Hauptachse einer Ellipse um den Mittelpunkt. Die Ellipse wird mit dw im Uhrzeigersinn um den Mittelpunkt x/y gedreht.
sw, ew	sw und ew geben den Start- und Endwinkel des Ellipsenbogens an.
s	gibt den Schrittwinkel an, mit dem die einzelnen Bogenpunkte errechnet werden.

Der Befehl errechnet einzelne Punkte des Ellipsenbogens, die mit einer Linie verbunden werden. Die Winkel müssen im Bogenmaß angegeben werden. Die Drehungen werden mit dem Uhrzeigersinn ausgeführt.
CELLIPSE x,y,a,b,dw,sw,ew,s C'E
Wie SELLIPSE, der Bogen wird jedoch gelöscht.
IELLIPSE x,y,a,b,dw,sw,ew,s I'E
Wie SELLIPSE, der Bogen wird jedoch invertiert.
STEXT r,b,h,v,a,x,y,a$ STE'X
Schreibt einen Text in den Grafikschirm.
Die einzelnen Parameter bedeuten:

r	Schreibrichtung r = 1 von links nach rechts r = 2 von rechts nach links r = 3 von unten nach oben r = 4 von oben nach unten Die Schrift ist, der Richtung entsprechend, gekippt.
b	Breite eines einzelnen Zeichens; von 1 (normal) bis 25 möglich.
h	Höhe eines Zeichens (von 1 bis 25).
v	Verschiebung (Kursivschrift) v gibt die Verschiebung eines Zeichens von seiner Spitze zum Fuß an. v kann Werte von 0 (normal) bis zum achtfachen (45 Grad Schräge) der Höhe h annehmen.
a	Abstand der Zeichen. a kann Werte von 0 bis 199 annehmen. Für Normalschrift ist a=8.
x und y	Geben die Startkoordinaten an.
x/y	ist die linke obere Ecke des ersten Zeichens eines Strings.
a$	String- oder numerischer Ausdruck, der gezeichnet werden soll. Im String kann zwischen den Zeichensätzen umgeschaltet werden: ctrl/9 rvs on ctrl/0 rvs off ctrl/a Groß/Klein ctrl/b Klein/Groß Ohne Umschaltung wird mit dem Groß/Grafik-Zeichensatz in Normalschrift gearbeitet.
Hinweis:	Wenn ein String außerhalb des Bildschirmes und mit großen Buchstaben gezeichnet wird, kann es vorkommen, daß der Computer bis zu zwei Minuten lang keine Meldung ausgibt. Er ist nicht abgestürzt, sondern fährt anschließend mit der Programmausführung fort.

CTEXT r,b,h,v,a,x,y,a$ C'T
Wie STEXT, der Text wird jedoch gelöscht.
ITEXT r,b,h,v,a,x,y,a$ I'T
Wie STEXT, jedoch wird der Text invertiert.
FILL x,y F'I
Malt ein durch Linien oder den Bildschirmrand begrenztes Feld beliebiger Form aus.
x/y geben die Startkoordinaten an.
DUPLICATE x0,y0,x1,y1,x,y D'U
Kopiert einen Grafikausschnitt.
Dabei geben x0/y0 und x1/y1 die Eckpunkte (links oben / rechts unten) eines Rechtecks an, das nun so kopiert wird, daß x/y die Koordinaten der linken oberen Ecke der Kopie angeben.
Der Ausschnitt wird so kopiert, daß die Kopie exakt dem Original entspricht. Das Original kann also überschrieben werden.
SCROLL r,x0,y0,x1,y1 SC'R
Rolliert einen Grafikausschnitt um einen Punkt.
Der Ausschnitt wird mit x0/y0 und x1/y1 analog zu DUPLICATE definiert.
r bestimmt die Richtung des Rollierens:
r1 = nach rechts
r2 = nach links
r3 = nach oben
r4 = nach unten
WINDOW y0,y1 W'I
WINDOW ermöglicht die gleichzeitige Anzeige von Text und Grafik. Die beiden Übergangszeilen dazu sind frei wählbar.
Die Parameter geben die erste (y0) und letzte (y1) Zeile im Grafikmodus an. Der restliche Bildschirm wird im Textmodus ausgegeben. In den Grenzzeilen sollten keine Textzeichen stehen; der VIC kann diese unter Umständen nicht von der Grafik unterscheiden und unsinnige Zeichen anzeigen.
WINDOW hat Vorrang vor MODE n.
WINDOW y0,y1 wird durch WINDOW ohne Parameter abgeschaltet. Der Groß/Grafik-Textmodus wird eingeschaltet.
LOWCOL x,y,p,h LO'W
Mit LOWCOL kann man gezielt ein Byte des Video-RAMs ändern.
x/y geben die Koordinaten des entsprechenden Grafikpunktes an, p die Punkt- und h die Hintergrundfarbe.
SPRITE n,x,y SP'R
Mit Sprite kann man die Form eines Sprites anhand der Grafik definieren.
n gibt die Nummer des Sprites an (0-7) und x/y die Koordinaten der linken oberen Ecke eines Grafikausschnittes, nach dem die Spriteform gebildet wird (siehe auch DUPLICATE).
Diese Form wird jedoch nicht einem Sprite zugeordnet; die Nummer n zeigt vielmehr die Blocknummer an, in den die Spriteform abgelegt wird. Dies sind die Blöcke 120 (n=0) bis 127 (n=7).
Die Blockzeiger liegen in den Adressen 50168 bis 50175 und müssen mit POKE gesetzt werden.
Hinweis: Die Spriteformen gelten nur im Grafikmodus. Im Textmodus gelten die gewohnten Regeln zum Arbeiten mit Sprites von Basic aus.
Das bedeutet, daß man mit WINDOW »gemischte« Sprites erzeugen und zwischen den zwei Definitionssätzen umschalten kann.
SSAVE n,n$,dv S'S
Speichert einen Spriteblock mit dem Namen n$ auf dv ab.
Zur Nummer n (0-7) siehe auch SPRITE.
SLOAD n,n$,dv SL'O
Lädt den Spriteblock n. Die Nummer, unter der der Block abgespeichert wurde, ist irrelevant. Siehe auch SPRITE und GLOAD.
Grafik 2000 besitzt vier weitere Befehle, die die meisten übrigen Befehle verändern und diese somit auf 140 verschiedene Funktionen erweitern.
PSCREEN n PSC'R

n = 2	Alle Befehle, die die Grafik verändern, wirken nicht mehr auf die sichtbare, sondern auf die unsichtbare, zweite Grafik. (CLEAR,COMBI,HLINE,INVERS,TEXT,LINE, CIRCLE,FILL, etc.)

n = 1 Die Befehle wirken wieder auf die sichtbare Grafik.

TSCREEN n **T'S**

n = 2 Die Befehle SPRITE, SCROLL und DUPLICATE sowie die Funktionen TEST und CTEST testen nicht in der sichtbaren, sondern in der unsichtbaren Grafik.
Man kann also mit DUPLICATE Ausschnitte der einen in die andere Grafik übertragen.

n = 1 schaltet zurück in die sichtbare Grafik.

COLPLOT p,h **COL'P**

COLPLOT ermöglicht das mehrfarbige Zeichnen in der HiRes-Grafik.
Parameter:
p Punktfarbe (0-15)
h Hintergrundfarbe (0-15)

COLPLOT p,h bewirkt, daß die Befehle

SPOINT	SLINE	SCIRCLE	SELLIPSE
STEXT	FILL	SCROLL	DUPLICATE

von nun an das Video-RAM mit der Punktfarbe p setzen. Jeder gesetzte Punkt erhält also die Farbe p; die Farbe der nicht gesetzten Punkte bleibt erhalten.

COLPLOT bewirkt außerdem, daß die Befehle

CPOINT	CLINE	CCIRCLE	CELLIPSE
CTEXT	SCROLL	DUPLICATE	

das Video-RAM mit der Hintergrundfarbe setzen.
Jeder gelöschte Punkt erhält die Farbe h; die Farbe der gesetzten Punkte wird nicht verändert.
Die Befehle SCROLL und DUPLICATE setzen sowohl Hintergrund- als auch Punktfarbe, je nachdem, ob ein Punkt gesetzt oder gelöscht wird.

Die Option kann mit COLPLOT ohne Parameter abgeschaltet werden.

PLOT n **P'L**

n = 2
Die Befehle

SPOINT	SLINE	SCIRCLE	SELLIPSE
STEXT	SCROLL	DUPLICATE	

arbeiten nicht mehr mit der normalen Punkt-Setzroutine, sondern mit der PSCLINE-Routine.
Es wird also unter jedem gezeichnetem Punkt eine Linie bis zum Bildschirmrand gelöscht.
Es lassen sich auf einfachste Weise effektvolle 3D-Bilder auf den Bildschirm zaubern.
Dieser Befehl arbeitet unabhängig von COLPLOT.
Es wird nur die Farbe des gezeichneten Punktes gesetzt; die Löschlinie verändert keinesfalls das Video-RAM.

n = 1 Schaltet wieder die normale Punktsetzroutine ein.

Hinweis: Die invertierten Befehle werden weder von COLPLOT noch von PLOT beeinflußt.

Testfunktionen

var = TEST (x,y) **T'E**

Weist der Variablen var den Wert 1 zu, wenn der Punkt x/y gesetzt ist, ansonsten den Wert 0.

var = CTEST (x,y) **CTE'S**

Weist der Variablen var die sichtbare Farbe des Punktes x/y zu (0 bis 15).
Siehe auch TSCREEN.

Sonstiges

Grafik 2000 stellt dem Benutzer eine verbesserte SQR-Routine zur Verfügung. Sie wurde bereits in den Interpreter eingebunden.

Wenn eine Basic-Fehlermeldung ausgegeben wird, schaltet Grafik 2000 den Textmodus ein, und der Befehl WINDOW wird abgeschaltet.

Grafik 2000 verfügt über eine eigenen NMI/BREAK-Routine.

Diese schaltet den Textmodus ein, den Befehl WINDOW ab und setzt Rahmen/Hintergrund- und Schriftfarben.

Diese Farbe kann man durch POKE selbst bestimmen:
POKE 51071,Hintergrundfarbe
POKE 51079,Schriftfarbe

Die Farben werden bereits beim Drücken der RESTORE-Taste ohne die RUN/STOP-Taste gesetzt.

Das Programm wird hierbei nicht unterbrochen.

Während den Zeichenbefehlen (ausgenommen ELLIPSE) kann man den Computer durch nichts in seiner Arbeit stören.

Auch RUN/STOP-RESTORE ist erst nach dem Ausführen eines solchen Befehls möglich.

(Frank-Rüdiger Brendel/ah)

```
programm : grafik 2000        0801 2298
----------------------------------------

0801 : 30 08 c1 07 9e 20 32 30  9b
0809 : 39 38 0d 0d 0d 0d 05 47  1f
0811 : 52 41 46 49 4b 20 32 30  9d
0819 : 30 30 20 42 59 20 46 52  06
0821 : 41 4e 4b 20 42 52 45 4e  c9
0829 : 44 45 4c 0d 0d 0d 00 00  fe
0831 : 00 78 a9 34 85 01 a0 1f  7f
0839 : b9 34 0c 99 14 03 88 10  de
0841 : f7 a9 16 8d 00 03 a9 c7  92
0849 : 8d 01 03 a0 07 b9 97 08  d8
0851 : 99 04 03 88 10 f7 c8 a9  f5
0859 : c4 85 15 84 14 a9 08 85  6f
0861 : fe a9 a0 85 fd a2 1a b1  cd
0869 : fd 91 14 88 d0 f9 e6 fe  bb
0871 : e6 15 ca d0 f2 a9 37 85  13
0879 : 01 20 ed c7 20 24 c9 20  89
0881 : ed c7 20 86 c7 20 22 e4  fb
0889 : a0 e0 b9 2d 07 20 d2 ff  6a
0891 : c8 d0 f7 4c 9d e3 00 c4  cb
0899 : 1a c5 bb c4 99 c6 00 a6  3a
08a1 : 7a a0 04 84 0f bd 00 02  e0
08a9 : 10 07 c9 ff f0 3e e8 d0  f5
08b1 : f4 c9 20 f0 37 85 08 c9  03
08b9 : 22 f0 56 24 0f 70 2d c9  2a
08c1 : 3f d0 04 a9 99 d0 25 c9  e7
08c9 : 30 90 04 c9 3c 90 1d 84  41
08d1 : 71 a0 00 84 0b 88 86 7a  27
08d9 : ca c8 e8 bd 00 02 38 f9  de
08e1 : 9e a0 f0 f5 c9 80 d0 30  0e
08e9 : 05 0b a4 71 e8 c8 99 fb  fe
08f1 : 01 b9 fb 01 f0 57 38 e9  6c
08f9 : 3a f0 04 c9 49 d0 02 85  14
0901 : 0f 38 e9 55 d0 9f 85 08  81
0909 : bd 00 02 f0 df c5 08 f0  93
0911 : db c8 99 fb 01 e8 d0 f0  b3
0919 : a6 7a e6 0b c8 b9 9d a0  29
0921 : 10 fa b9 9e a0 d0 b4 a0  95
0929 : ff ca c8 e8 bd 00 02 38  31
0931 : f9 a2 c5 f0 f5 c9 80 f0  9c
0939 : af a6 7a e6 0b c8 b9 a1  d8
0941 : c5 10 fa b9 a2 c5 d0 e4  69
0949 : bd 00 02 10 9d 99 fd 01  2a
0951 : c6 7b a9 ff 85 7a 60 4c  85
0959 : a5 a9 20 73 00 20 c4 c4  e7
0961 : 4c ae a7 f0 f0 e9 80 90  8e
0969 : ee c9 23 b0 11 c9 0b f0  88
0971 : 27 0a a8 b9 0d a0 48 b9  69
0979 : 0c a0 48 4c 73 00 c9 4c  68
0981 : 90 13 c9 75 b0 0f e9 4b  7e
0989 : 0a a8 b9 51 c5 48 b9 50  a6
0991 : c5 48 4c 73 00 4c 0e a8  e8
0999 : 20 73 00 20 9e ad 20 79  42
09a1 : 00 c9 89 f0 05 a9 a7 20  83
09a9 : ff ae a5 61 d0 03 4c 3b  62
09b1 : a9 20 79 00 b0 ad 4c a0  b4
09b9 : a8 c9 cc b0 05 29 ff 4c  c1
09c1 : 1a a7 c9 ff d0 03 4c f3  5f
09c9 : a6 24 0f 30 f9 38 e9 cb  ec
09d1 : aa 84 49 a0 ff ca f0 08  4e
09d9 : c8 b9 a2 c5 10 fa 30 f5  65
09e1 : c8 b9 a2 c5 10 03 4c ef  11
09e9 : a6 20 47 ab 4c 41 c5 ec  a6
09f1 : c7 1b c8 78 c8 3d c8 40  a1
09f9 : c8 43 c8 53 c8 5f c8 6c  83
0a01 : c8 cb c8 bc c8 ed c8 14  c0
0a09 : c9 23 c9 29 c9 3a c9 85  9c
0a11 : c9 88 c9 8b c9 7f ca 82  cb
0a19 : ca 85 ca 0f cb 12 cb 15  e1
0a21 : cb 83 cb 86 cb 89 cb 28  fa
0a29 : ce 5d c9 3e ce 70 ce 04  94
0a31 : ca cb ce 45 ca 41 cf 5c  ec
0a39 : cf 6a cf e2 cf 7c cf bd  29
0a41 : cf 43 4c 45 41 d2 43 4f  c4
0a49 : 4c 4f d2 4d 4f 44 c5 49  5c
0a51 : 50 4f 49 4e d4 43 50 4f  ac
0a59 : 49 4e d4 53 50 4f 49 4e  aa
0a61 : d4 50 53 43 4c 49 4e c5  6e
0a69 : 48 4c 49 4e c5 56 4c 49  c6
0a71 : 4e c5 48 4d 41 52 cb 56  e0
0a79 : 4d 41 52 cb 47 53 41 56  36
0a81 : c5 47 4c 4f 41 c4 43 48  bf
0a89 : 41 4e 47 c5 43 4f 4d c2  e5
0a91 : 4c 4f 57 43 4f cc 49 4c  dc
0a99 : 49 4e c5 43 4c 49 4e c5  b7
0aa1 : 53 4c 49 4e c5 49 54 45  b9
0aa9 : 58 d4 43 54 45 58 d4 53  d8
0ab1 : 54 45 58 d4 49 43 49 52  d1
0ab9 : 43 4c c5 43 43 49 52 43  4a
0ac1 : 4c c5 53 43 49 52 43 4c  fa
0ac9 : c5 49 45 4c 4c 49 50 53  05
0ad1 : c5 43 45 4c 4c 49 50 53  0a
0ad9 : c5 53 45 4c 4c 49 50 53  1a
0ae1 : c5 46 49 4c cc 49 4e 56  a2
0ae9 : 45 52 d3 53 43 52 4f 4c  53
0af1 : cc 44 55 50 4c 49 43 41  dd
0af9 : 54 c5 43 4f 4c 50 4c 4f  02
0b01 : d4 57 49 4e 44 4f d7 50  5c
0b09 : 4c 4f d4 54 53 43 52 45  e0
0b11 : 45 ce 50 53 43 52 45 45  a2
0b19 : ce 43 53 41 56 c5 53 50  07
0b21 : 52 49 54 c5 53 53 41 56  67
0b29 : c5 53 4c 4f 41 c4 54 45  ab
0b31 : 53 d4 43 54 45 53 d4 00  8c
0b39 : a9 00 85 0d 20 73 00 08  93
```

Listing zum Programm Grafik 2000. Bei der Eingabe bitte den MSE verwenden.

```
0b41 : c9 ba d0 0d 28 20 73 00  8e
0b49 : 20 f1 ae 20 93 cd 4c 8d  05
0b51 : ad c9 f5 f0 08 c9 f6 f0  0b
0b59 : 0b 28 4c 8d ae 28 20 ec  c4
0b61 : c6 4c a2 b3 28 20 ec c6  31
0b69 : a2 0f 98 f0 02 a2 f0 86  dd
0b71 : 02 a5 fe 38 e9 e0 20 d2  d8
0b79 : c7 a0 00 b1 fd 25 02 c9  6b
0b81 : 10 90 04 4a 4a 4a 4a a8  95
0b89 : 4c a2 b3 20 73 00 20 fa  c5
0b91 : ae a9 b2 48 a9 47 48 20  00
0b99 : 26 c7 68 68 86 02 20 f7  b3
0ba1 : ae 20 e1 c7 20 f2 cf 85  b4
0ba9 : 02 a0 00 b1 fd 25 02 f0  25
0bb1 : 01 c8 4c e7 c7 8a 30 0a  cc
0bb9 : 86 02 20 ce ce a6 02 4c  e5
0bc1 : 3a a4 4c 74 a4 20 eb b7  59
0bc9 : e0 c8 b0 0f a5 15 c9 01  47
0bd1 : 90 08 d0 07 a5 14 c9 40  1d
0bd9 : b0 01 60 68 68 60 a9 33  c6
0be1 : 85 01 b1 6a 85 fc a9 34  f0
0be9 : 85 01 60 48 8a 48 98 48  ee
0bf1 : a9 7f 8d 0d dd ac 0d dd  92
0bf9 : 10 03 4c 72 fe 20 bc f6  be
0c01 : 20 e1 ff d0 1a a0 1f b9  c2
0c09 : 94 c7 99 14 03 88 10 f7  0e
0c11 : 20 a3 fd 20 18 e5 20 86  c4
0c19 : c7 20 ce ce 6c 02 a0 20  17
0c21 : 86 c7 4c 72 fe a9 06 8d  5d
0c29 : 20 d0 8d 21 d0 a9 01 8d  b2
0c31 : 86 02 60 31 ea 66 c7 4c  90
0c39 : c7 4a f3 91 f2 0e f2 50  60
0c41 : f2 33 f3 57 f1 ca f1 ed  ce
0c49 : f6 3e f1 2f f3 66 fe a5  7a
0c51 : f4 ed f5 a9 fc 8d fa ff  16
0c59 : 8d fa bf a9 ff 8d fb ff  e5
0c61 : 8d fb bf a9 40 8d fc ff  75
0c69 : 8d fc bf 60 a5 fe 38 e9  77
0c71 : e0 4a 66 fd 4a 66 fd 4a  34
0c79 : 66 fd 18 69 c0 85 fe 60  06
0c81 : 78 a9 34 85 01 60 a9 37  b4
0c89 : 85 01 58 60 a9 e0 85 fe  67
0c91 : a9 00 85 fd a8 a2 20 91  9f
0c99 : fd c8 d0 fb e6 fe ca d0  e1
0ca1 : f6 4c b4 c7 20 9e b7 8a  ce
0ca9 : c9 10 b0 75 0a 0a 0a 0a  82
0cb1 : 85 02 20 f1 b7 8a c9 10  95
0cb9 : b0 67 60 20 05 c8 05 02  e8
0cc1 : a2 c0 86 fe a0 00 84 fd  5d
0cc9 : a2 03 91 fd c8 d0 fb e6  e2
0cd1 : fe ca d0 f6 a2 e8 9d ff  2f
0cd9 : c2 ca d0 fa 60 a9 c0 2c  42
0ce1 : a9 80 2c a9 00 85 09 20  9b
0ce9 : 26 c7 20 e1 c7 20 4f d0  93
0cf1 : 4c e7 c7 20 26 c7 20 e1  0c
0cf9 : c7 20 a7 d0 4c e7 c7 20  38
0d01 : 9e b7 e0 c8 b0 f6 20 e1  d3
0d09 : c7 4c 94 d1 20 8a ad 20  a3
0d11 : f7 b7 20 e1 c7 4c ca d1  d6
0d19 : 20 9e b7 e0 01 90 21 f0  8d
0d21 : 03 4c 48 b2 78 ad 18 d0  aa
0d29 : 8d a3 c8 a9 0b a0 3b a2  d7
0d31 : 94 8d 18 d0 8c 11 d0 8e  5e
0d39 : 00 dd a9 60 a2 78 d0 14  f8
0d41 : 78 a9 15 a0 1b a2 97 8d  27
0d49 : 18 d0 8c 11 d0 8e 00 dd  4c
0d51 : a9 78 a2 60 8d 85 c8 8e  30
0d59 : a1 c8 58 60 20 26 c7 ca  68
0d61 : ca e0 c8 b0 97 20 e1 c7  75
0d69 : 4c fe d1 20 26 c7 a5 14  0c
0d71 : 38 e9 02 85 14 a5 15 e9  66
0d79 : 00 85 15 c9 01 90 08 d0  11
0d81 : 24 a5 14 c9 40 b0 d4 20  d3
0d89 : e1 c7 4c 47 d2 20 b4 c7  da
0d91 : 20 09 c9 78 a2 36 86 01  af
0d99 : a2 a0 86 fe a2 00 86 fd  4d
0da1 : a0 c0 20 d8 ff 4c e7 c7  56
0da9 : 20 d4 e1 a5 b9 09 01 85  53
0db1 : b9 a9 fd 60 20 d4 e1 a9  4e
0db9 : 01 85 b9 a9 00 20 9e f4  86
0dc1 : 4c b4 c7 20 e1 c7 4c ee  c9
0dc9 : dc 20 9e b7 ca e0 03 90  35
0dd1 : 03 4c 48 b2 20 e1 c7 4c  2b
0dd9 : 1f dd 20 26 c7 86 08 20  c5
0de1 : fd ae 20 05 c8 05 02 85  a6
0de9 : 92 20 e1 c7 a6 08 20 11  4a
0df1 : d0 20 cd c7 a0 00 a5 92  03
0df9 : 91 fd 4c e7 c7 20 b4 c7  79
0e01 : 20 e1 c7 20 0b dd ca b1  36
0e09 : fd 49 ff 91 fd c8 d0 f7  36
0e11 : e6 fe ca d0 f2 a0 41 b9  f0
0e19 : ff fe 49 ff 99 ff fe 88  90
0e21 : d0 f5 4c e7 c7 a9 c0 2c  21
0e29 : a9 80 2c a9 00 85 09 20  e3
0e31 : 26 c7 86 02 20 e1 c7 a5  98
0e39 : 14 8d ee dd a5 15 8d ef  a4
0e41 : dd 20 79 00 d0 1e ad f6  2f
0e49 : dd 85 02 8e f0 dd 8e f6  61
0e51 : dd a2 01 bd f4 dd 95 14  34
0e59 : bd ee dd 9d f4 dd ca 10  42
0e61 : f2 4c e9 c9 20 e7 c7 20  cd
0e69 : fd ae 20 26 c7 20 e1 c7  1f
0e71 : 8e f0 dd 8e f6 dd a2 01  ab
0e79 : b5 14 bc ee dd 9d ee dd  87
0e81 : 9d f4 dd 94 14 ca 10 f0  5c
0e89 : ae f0 dd 20 73 d2 4c e7  f9
0e91 : c7 20 9e b7 ca 8a 30 78  ba
0e99 : c9 04 b0 74 20 e1 c7 8a  64
0ea1 : 6a 6a 6a 60 d0 15 20 e1  e1
0ea9 : c7 a9 49 8d 5a d0 a9 ff  1c
0eb1 : 8d 5b d0 a2 0a a0 4c a9  9e
0eb9 : 60 d0 1e 20 05 c8 20 e1  e8
0ec1 : c7 8e 4b d0 a5 02 8d 73  44
0ec9 : d0 a9 50 8d 5a d0 a9 e1  ca
0ed1 : 8d 5b d0 a2 0f a0 20 a9  5e
0ed9 : ea 8d 66 da 8e 57 d0 8c  7f
0ee1 : 79 d0 4c e7 c7 20 9e b7  3a
0ee9 : e0 01 f0 0f e0 02 d0 20  09
0ef1 : 20 e1 c7 a2 2c a0 30 a9  24
0ef9 : 48 d0 09 20 e1 c7 a2 20  17
0f01 : a0 a0 a9 00 8e 4f d0 8c  1b
0f09 : 52 d0 8d 53 d0 4c e7 c7  30
0f11 : 4c 48 b2 20 f1 b7 8a 38  a9
0f19 : f0 f6 c9 1a b0 f2 60 a9  b1
0f21 : c0 2c a9 80 2c a9 00 85  8d
0f29 : 09 20 f2 c9 20 6b d1 20  1d
0f31 : e7 c7 20 74 ca 85 92 20  f6
0f39 : 74 ca 85 93 20 f1 b7 a5  a2
0f41 : 93 0a 0a 0a c5 65 90 c8  f9
0f49 : 86 ae 20 f1 b7 e0 c9 b0  78
0f51 : bf 86 9c 20 fd ae 20 26  a1
0f59 : c7 86 02 a9 d0 85 fb a9  96
0f61 : 00 85 a4 85 a5 85 11 20  09
0f69 : fd ae 20 9e ad 24 0d 30  2a
0f71 : 06 20 dd bd 20 87 b4 20  08
0f79 : a6 b6 48 a5 9c 4a 85 9e  b1
0f81 : a2 00 a9 a0 38 e5 9e e8  a1
0f89 : f0 08 c9 01 b0 f7 8a 69  d8
0f91 : 0b aa ca 86 9e 68 c5 9e  f6
0f99 : b0 01 aa a0 00 e8 20 e1  14
0fa1 : c7 ca f0 08 b1 22 20 0e  d3
0fa9 : d5 c8 d0 f5 4c e7 c7 a9  4c
0fb1 : c0 2c a9 80 2c a9 00 85  1d
0fb9 : 09 20 26 c7 86 ae a5 15  f3
0fc1 : 85 93 a5 14 85 92 20 fd  65
0fc9 : ae 20 eb b7 a5 15 d0 dc  79
0fd1 : a5 14 85 5a 86 5b 18 65  9b
0fd9 : 92 aa a5 93 69 00 c9 01  5c
0fe1 : 90 06 d0 c8 e0 40 b0 c4  1e
0fe9 : a5 92 38 e5 5a a5 93 e9  97
0ff1 : 00 90 b9 a5 ae aa 18 65  c8
0ff9 : 5b b0 b1 c9 c8 b0 ad 8a  30
1001 : 38 e5 5b 90 a7 20 e1 c7  a7
1009 : 4c 0f d7 20 d4 bb 20 [illegible]  7e
1011 : ae 20 8a ad a5 61 60 20  4f
1019 : 6f cb f0 04 c9 8c 90 f6  5c
1021 : 4c 48 b2 a9 c0 2c a9 80  88
1029 : 2c a9 00 85 09 20 72 cb  ce
1031 : 20 7d cb 20 9b bc a5 65  08
1039 : 8d 25 ce a5 64 8d 26 ce  aa
1041 : 20 6f cb 20 7d cb 20 9b  fe
1049 : bc a5 65 8d 27 ce a5 64  2b
1051 : 8d 28 ce 20 78 cb a2 ef  fa
1059 : a0 cd 20 d4 bb 20 78 cb  b9
1061 : a2 f4 a0 cd 20 6c cb a2  39
1069 : 03 a0 ce 20 6c cb a2 0d  3e
1071 : a0 ce 20 6c cb a2 f9 a0  09
1079 : cd 20 6c cb f0 a2 a2 fe  97
1081 : a0 cd 20 d4 bb a9 03 a0  01
1089 : ce 20 a2 bb 20 6b e2 a2  b6
1091 : 08 a0 ce 20 d4 bb a9 03  79
1099 : a0 ce 20 a2 bb 20 64 e2  11
10a1 : a2 03 a0 ce 20 d4 bb 20  9f
10a9 : 9b cc a5 ae 8d 23 ce a5  62
10b1 : af 8d 24 ce a5 92 8d 21  71
10b9 : ce a5 93 8d 22 ce 20 9b  41
10c1 : cc 20 e1 c7 a5 af d0 48  ba
10c9 : a5 ae c9 c8 b0 42 8d f0  86
10d1 : dd aa a4 92 a5 93 c9 01  9f
10d9 : 90 06 d0 34 c0 40 b0 30  58
10e1 : 8c ee dd 8d ef dd ad 24  fa
10e9 : ce d0 25 ad 23 ce c9 c8  80
10f1 : b0 1e 85 02 ac 21 ce ad  bc
10f9 : 22 ce c9 01 90 06 d0 10  b1
1101 : c0 40 b0 0c 84 14 85 15  b8
1109 : 20 73 d2 a6 02 20 4f d0  6c
1111 : 20 e7 c7 a9 0d a0 ce 20  9d
1119 : a2 bb a9 fe a0 cd 20 67  ab
1121 : b8 a2 0d a0 ce 20 d4 bb  3a
1129 : 20 1b bc a9 f9 a0 cd 20  57
1131 : 5b bc aa ca f0 03 4c 0b  5d
1139 : cc 60 a9 0d a0 ce 20 a2  88
1141 : bb 20 6b e2 a2 12 a0 ce  1e
1149 : 20 d4 bb a9 0d a0 ce 20  49
1151 : a2 bb 20 64 e2 a5 61 d0  e8
1159 : 1d a2 1c a0 ce 20 d4 bb  9b
1161 : a9 ef a0 cd 20 a2 bb ad  45
1169 : 13 ce 85 66 a2 17 a0 ce  14
1171 : 20 d4 bb 4c 3b cd 20 2b  6d
1179 : bc 85 0a a9 12 a0 ce 20  51
1181 : 0f bb a2 12 a0 ce 20 d4  03
1189 : bb 20 dc cd 20 e9 cd a5  19
1191 : 61 20 2b ba 20 ca bb a9  bf
1199 : ef a0 cd 20 a2 bb 20 e9  ac
11a1 : cd a5 61 20 2b ba a9 57  7b
11a9 : a0 00 20 67 b8 20 93 cd  b5
11b1 : a9 bc a0 b9 20 0f bb a9  d4
11b9 : ef a0 cd 20 28 ba 20 dc  02
11c1 : cd a5 0a 85 66 a2 1c a0  c1
11c9 : ce 20 d4 bb a9 12 a0 ce  9f
11d1 : 20 28 ba a2 17 a0 ce 20  fa
11d9 : d4 bb 20 cf cd 20 ca bb  0d
11e1 : 20 e2 cd 20 d6 cd a9 57  1b
11e9 : a0 00 20 67 b8 20 49 b8  a1
11f1 : 20 9b bc 18 a5 65 6d 25  96
11f9 : ce 85 92 a5 64 6d 26 ce  cb
1201 : 85 93 a9 17 a0 ce 20 a2  e3
1209 : bb 20 d6 cd 20 ca bb 20  cb
1211 : e2 cd 20 cf cd a9 57 a0  a5
1219 : 00 20 50 b8 20 49 b8 20  c4
1221 : 9b bc 18 a5 65 6d 27 ce  d1
1229 : 85 ae a5 64 6d 28 ce 85  5a
1231 : af 60 20 2b bc f0 fa 10  dd
1239 : 03 4c 48 b2 20 ca bb a5  5d
1241 : 61 38 e9 81 08 4a 18 69  6f
1249 : 01 28 90 02 69 7f 85 61  2e
1251 : a9 04 85 67 20 c7 bb a9  cd
1259 : 57 a0 00 20 0f bb a9 5c  33
1261 : a0 00 20 67 b8 c6 61 c6  cb
1269 : 67 d0 e9 a5 61 60 a9 03  2d
1271 : a0 ce 4c 28 ba a9 08 a0  eb
1279 : ce d0 f7 a9 f4 a0 cd d0  0f
1281 : f1 a9 1c a0 ce 4c a2 bb  b3
1289 : 20 0c bc 86 6f 60 00 00  a9
1291 : 00 00 00 00 00 00 00 00  92
1299 : 00 00 00 00 00 00 00 00  9a
12a1 : 00 00 00 00 00 00 00 00  a2
12a9 : 00 00 00 00 00 00 00 00  aa
12b1 : 00 00 00 00 00 00 00 00  b2
12b9 : 00 00 00 00 00 00 00 00  ba
12c1 : 00 00 00 00 00 00 00 00  c2
12c9 : 20 26 c7 86 63 a5 14 85  7e
12d1 : 61 a5 15 85 62 20 e1 c7  39
12d9 : 20 84 d8 4c e7 c7 20 f2  1e
12e1 : c9 10 02 49 40 85 09 20  f1
12e9 : 0b d1 20 fd ae 20 26 c7  b9
12f1 : 86 9c a5 14 85 92 a5 15  5f
12f9 : 85 93 20 fd ae 20 26 c7  24
1301 : 86 fb a5 14 85 a4 a5 15  af
1309 : 85 a5 20 e1 c7 4c a2 da  c4
1311 : 20 26 c7 86 9c a5 14 85  59
1319 : a4 a5 15 85 a5 20 fd ae  37
1321 : 20 26 c7 86 fb a5 14 85  5f
1329 : 92 a5 15 85 93 20 fd ae  13
1331 : 20 26 c7 86 63 a5 14 85  e6
1339 : 61 a5 15 85 62 20 e1 c7  a1
1341 : 4c 4d db ad 19 d0 8d 19  61
1349 : d0 30 07 ad 0d dc 58 4c  5b
1351 : 31 ea ad 12 d0 c9 ff b0  62
1359 : 0b 20 85 c8 a9 ff 8d 12  e4
1361 : d0 4c 81 ea 20 a1 c8 a9  9a
1369 : ff d0 f3 d0 0d 78 a9 31  85
1371 : a0 ea a2 00 20 2d cf 4c  72
1379 : a1 c8 20 9e b7 8a c9 c8  e3
1381 : 90 03 4c 48 b2 8d 15 cf  3a
1389 : 20 f1 b7 8a c9 c8 b0 f2  6c
1391 : 78 cd 15 cf b0 06 a2 a1  38
1399 : a0 85 d0 0b ae 15 cf 8d  7f
13a1 : 15 cf 8a a2 85 a0 a1 8e  96
13a9 : bb ce 8c c6 ce 18 69 31  7d
13b1 : 8d c9 ce a9 ff 69 33 8d  3f
13b9 : be ce 8d b7 ce 8d 12 d0  7c
13c1 : 2e 11 d0 4e 11 d0 a2 81  9b
13c9 : a9 a4 a0 ce 8c 15 03 8d  5f
13d1 : 14 03 8e 1a d0 58 60 20  df
13d9 : 9e b7 e0 08 b0 a4 4c fd  e9
13e1 : ae 20 9e b7 ca e0 01 f0  d7
13e9 : 05 b0 97 a2 e0 2c a2 a0  bc
13f1 : 20 e1 c7 8e d0 8e d6 15
13f9 : c6 4c e7 c7 20 9e b7 ca  44
1401 : e0 02 b0 e5 20 e1 c7 4c  94
1409 : 65 dc 20 09 c9 a2 00 86  c4
1411 : fd a2 c0 86 fe a2 e9 a0  4e
1419 : c3 4c d8 ff 20 38 cf 20  7c
1421 : a9 cf a0 20 09 c9 20 e1 c7  90
1429 : a0 3f b1 fd 99 40 bf 88  40
1431 : 10 f8 a2 80 a0 bf a9 40  a5
1439 : 85 fd 84 fe a9 36 85 01  22
1441 : a9 fd 20 d8 ff 4c e7 c7  9d
1449 : a0 de a9 00 e8 ca f0 08  7b
1451 : 18 69 40 90 f8 c8 b0 f5  c4
1459 : 85 fd 84 fe 60 20 38 cf  65
1461 : 86 fb 20 d4 e1 a9 01 85  02
1469 : b9 a9 00 20 9e f4 20 e1  d1
1471 : c7 a6 fb 20 a9 cf a0 3f  a8
1479 : b9 40 bf 91 fd 88 10 f8  cb
1481 : 30 c3 20 38 cf 20 a9 cf  e6
1489 : 20 26 c7 20 e1 c7 4c 95  6b
1491 : dc 8a 29 07 85 fd 8a 29  a2
1499 : f8 4a 4a a8 b9 4d dd 85  e6
14a1 : fe a5 14 29 f8 65 fd 79  42
14a9 : 4c dd 85 fd a9 e0 d0 1d  24
14b1 : 8a 29 07 85 fd 8a 29 f8  0d
```

Listing zum Programm Grafik 2000 (Fortsetzung)

```
14b9 : 4a 4a a8 b9 4d dd 85 fe  61
14c1 : a5 14 29 f8 65 fd 79 4c  9e
14c9 : dd 85 fd a9 e0 65 15 65  76
14d1 : fe 85 fe a5 14 29 07 aa  02
14d9 : bd 7e dd 60 49 ff 31 fd  ae
14e1 : 91 fd 20 cd c7 b1 fd 29  87
14e9 : f0 09 0f 91 fd 60 20 11  d9
14f1 : d0 a0 00 24 09 10 0a 70  b0
14f9 : 05 49 ff 31 fd 2c 51 fd  4b
1501 : 2c 11 fd 91 fd 60 11 fd  8a
1509 : 91 fd 20 cd c7 b1 fd 29  af
1511 : 0f 09 f0 91 fd 60 86 02  14
1519 : 4c a7 d0 a5 02 29 f8 4a  03
1521 : 4a a8 a5 15 4a a5 14 6a  c2
1529 : 4a 4a 18 79 92 dd 85 fd  f8
1531 : b9 93 dd 69 c0 85 fe a0  ce
1539 : 00 f0 d2 24 09 10 d7 20  9b
1541 : 11 d0 a0 00 f0 ad 8a 4a  1e
1549 : 4a 4a 48 0a a8 b9 4d dd  55
1551 : 85 fe a5 14 29 f8 79 4c  1a
1559 : dd 85 fd 8a 29 07 a8 20  77
1561 : 2c d0 aa 11 fd 91 fd 8a  3c
1569 : 49 ff aa 8a c8 c0 08 b0  c2
1571 : 06 31 fd 91 fd 90 f4 68  ca
1579 : 69 00 c9 19 b0 97 48 a5  ac
1581 : fd 69 40 85 fd a5 fe 69  cf
1589 : 01 85 fe a0 ff d0 dc 2a  6f
1591 : b0 0c 30 05 a2 40 a0 3b  19
1599 : 60 a2 47 a0 49 60 30 05  93
15a1 : a2 50 a0 40 60 a2 57 a0  55
15a9 : 4e 60 a5 09 29 80 20 f0  ab
15b1 : d0 6a 8c c5 da 09 40 20  c9
15b9 : f0 d0 6a 8c d4 da 49 80  88
15c1 : 20 f0 d0 6a 8e ce da 8c  bf
15c9 : f8 da 29 80 20 f0 d0 8e  73
15d1 : d1 da 8c e4 da a5 09 20  0e
15d9 : f0 d0 8e f2 da 8e 05 db  21
15e1 : 6a 49 40 20 f0 d0 8e f5  c0
15e9 : da 8e fd da 4c e7 c7 2a  5c
15f1 : b0 0c 30 05 a2 60 a0 47  92
15f9 : 60 a2 6a a0 40 60 30 05  2b
1601 : a2 80 a0 50 60 a2 76 a0  4c
1609 : 57 60 20 50 d1 6a 8e 86  5a
1611 : d5 8c e4 d5 8e 3e d6 49  e9
1619 : 40 20 50 d1 6a 8e 21 d6  05
1621 : 8c 11 d6 49 80 10 02 49  38
1629 : 40 20 50 d1 8c 24 d6 8e  2a
1631 : 35 d6 60 8a 29 f8 4a 4a  53
1639 : a8 b9 4c dd 85 fd 8a 29  51
1641 : 07 65 fd 85 fd b9 4d dd  c9
1649 : 69 e0 85 fe a0 00 a9 ff  14
1651 : 91 fd 98 69 08 a8 90 f6  2a
1659 : e6 fe a0 38 a9 ff 91 fd  ca
1661 : 98 e9 08 a8 b0 f6 4c e7  c9
1669 : c7 a5 14 aa 29 f8 85 fd  ca
1671 : a5 15 18 69 e0 85 fe 8a  1f
1679 : 20 36 d0 aa a9 19 85 09  c9
1681 : a0 07 8a 11 fd 91 fd 88  df
1689 : 10 f8 a5 fd 18 69 40 85  17
1691 : fd a5 fe 69 01 85 fe c6  13
1699 : 09 d0 e5 f0 c9 8a 4a 4a  50
16a1 : 4a 48 0a a8 b9 4d dd 85  2f
16a9 : fe a5 14 29 f8 79 4c dd  ec
16b1 : 85 fd 8a 29 07 a8 20 2c  8b
16b9 : d0 85 09 a2 05 a5 09 11  a6
16c1 : fd 91 fd ca f0 1d c8 c0  fc
16c9 : 08 d0 f2 68 69 00 c9 19  f3
16d1 : b0 94 48 a5 fd 69 40 85  c9
16d9 : fd a5 fe 69 01 85 fe a0  0f
16e1 : 00 f0 da 68 90 29 a5 15  30
16e9 : 6a a5 14 6a 4a 4a 85 09  97
16f1 : 20 11 d0 a0 00 bd 86 dd  a6
16f9 : 11 fd 91 fd e0 04 90 0f  bb
1701 : a4 09 c0 27 b0 09 bd 8a  9e
1709 : dd a0 08 11 fd 91 fd 4c  58
1711 : e7 c7 e4 02 90 5a ad ef  c8
1719 : dd c5 15 90 09 d0 2c ad  53
1721 : ee dd c5 14 b0 25 20 b3  0e
1729 : d3 ad f2 dd d0 11 cc f1  f8
1731 : dd 90 0c 20 f5 d3 a9 d3  aa
1739 : 48 a9 96 48 4c bf d4 20  5b
1741 : e5 d3 a9 d3 48 a9 a6 48  f2
1749 : 4c 7f d4 20 b3 d3 ad f2  04
1751 : dd d0 11 cc f1 dd 90 0c  dd
1759 : 20 f5 d3 a9 d3 48 a9 30  24
1761 : 48 4c 3d d4 20 e5 d3 a9  8d
1769 : d3 48 a9 40 48 4c 7f d4  61
1771 : ad ef dd c5 15 90 09 d0  e1
1779 : 2c ad ee dd c5 14 b0 25  fd
1781 : 20 b3 d3 ad f2 dd d0 11  a9
1789 : cc f1 dd 90 0c 20 f5 d3  18
1791 : a9 d3 48 a9 52 48 4c bf  83
1799 : d4 20 e5 d3 a9 d3 48 a9  1f
17a1 : 62 48 4c 03 d5 20 b3 d3  70
17a9 : ad f2 dd d0 11 cc f1 dd  5c
17b1 : 90 0c 20 f5 d3 a9 d3 48  78
17b9 : a9 74 48 4c 3d d4 20 e5  ff
17c1 : d3 a9 d3 48 a9 84 48 4c  df
17c9 : 03 d5 20 0a d4 20 22 d4  81
17d1 : d0 07 ad f3 dd c5 5e b0  f5
17d9 : f1 60 20 45 d4 20 63 d4  31
17e1 : 90 f8 d0 07 ad f1 dd c5  70
17e9 : 5e b0 ef 60 20 8a d4 20  91
```

```
17f1 : a2 d4 d0 07 ad f3 dd c5  90
17f9 : 5e b0 f1 60 20 c7 d4 20  0c
1801 : e7 d4 90 f8 d0 07 ad f1  75
1809 : dd c5 5e b0 ef 60 20 8a  0e
1811 : d4 20 22 d4 d0 07 ad f3  fc
1819 : dd c5 5e b0 f1 60 20 45  b4
1821 : d4 20 e7 d4 90 f8 d0 07  bc
1829 : ad f1 dd c5 5e b0 ef 60  eb
1831 : 20 0a d4 20 a2 d4 d0 07  b2
1839 : ad f3 dd c5 5e b0 f1 60  04
1841 : 20 c7 d4 20 63 d4 90 f8  8f
1849 : d0 07 ad f1 dd c5 5e b0  2d
1851 : ef 60 8a 38 e5 02 b0 04  53
1859 : 49 ff 69 01 8d f3 dd a8  5e
1861 : a5 14 38 ed ee dd 8d f1  d4
1869 : dd a5 15 ed ef dd b0 10  ed
1871 : aa ad f1 dd 49 ff 69 01  66
1879 : 8d f1 dd 8a 49 ff 69 00  02
1881 : 8d f2 dd 60 20 01 d4 ad  c3
1889 : f2 dd 4a 85 5d ad f1 dd  74
1891 : 6a 85 5c 60 20 01 d4 85  49
1899 : 5d ad f3 dd 4a 85 5c 60  88
18a1 : a9 01 85 5e a9 00 85 5f  67
18a9 : 60 e6 02 e6 5e d0 02 e6  1c
18b1 : 5f a5 5c 18 6d f1 dd 85  e6
18b9 : 5c a5 5d 6d f2 dd 85 5d  dc
18c1 : 60 d0 07 ad f3 dd c5 5c  ff
18c9 : b0 12 a5 5c 38 ed f3 dd  f6
18d1 : 85 5c b0 02 c6 5d e6 14  0c
18d9 : d0 02 e6 15 a6 02 20 4f  a0
18e1 : d0 a5 5f 60 e6 14 d0 02  be
18e9 : e6 15 e6 5e d0 02 e6 5f  57
18f1 : a5 5c 18 6d f3 dd 85 5c  75
18f9 : a5 5d 69 00 85 5d cd f2  08
1901 : dd 60 90 1a d0 07 ad f1  56
1909 : dd c5 5c b0 11 a5 5c 38  16
1911 : ed f1 dd 85 5c a5 5d ed  63
1919 : f2 dd 85 5d e6 02 a6 02  24
1921 : 20 4f d0 a5 5f cd f2 dd  bd
1929 : 60 c6 02 e6 5e d0 02 e6  8c
1931 : 5f a5 5c 18 6d f1 dd 85  66
1939 : 5c a5 5d 6d f2 dd 85 5d  5c
1941 : 60 d0 07 ad f3 dd c5 5c  7f
1949 : b0 14 a5 5c 38 ed f3 dd  77
1951 : 85 5c b0 02 c6 5d a5 14  87
1959 : d0 02 c6 15 c6 14 a6 02  2a
1961 : 20 4f d0 a5 5f 60 a5 14  c9
1969 : d0 02 c6 15 c6 14 e6 5e  f4
1971 : d0 02 e6 5f a5 5c 18 6d  60
1979 : f3 dd 85 5c a5 5d 69 00  33
1981 : 85 5d cd f2 dd [illegible] 90 1a  de
1989 : d0 07 ad f1 dd c5 5c b0  65
1991 : 11 a5 5c 38 ed f1 dd 85  84
1999 : 5c a5 5d ed f2 dd 85 5d  cc
19a1 : c6 02 a6 02 20 4f d0 a5  5d
19a9 : 5f cd f2 dd 60 c9 12 d0  a5
19b1 : 0e 66 a4 24 a5 30 03 a9  db
19b9 : d4 2c a9 dc 85 fb 60 c9  f6
19c1 : 92 d0 0e 46 a4 24 a5 30  6a
19c9 : 03 a9 d0 2c a9 d8 85 fb  ca
19d1 : 60 c9 01 d0 08 46 a5 24  02
19d9 : a4 10 ee 30 da c9 02 d0  ed
19e1 : 08 66 a5 24 a4 10 e5 30  cd
19e9 : d1 c9 ff f0 2d 29 ff 48  69
19f1 : 10 17 0a 10 0f 0a 10 07  a1
19f9 : 68 09 40 29 7f d0 1d 68  df
1a01 : 29 5f d0 18 0a 30 f1 68  cc
1a09 : 60 0a 10 09 0a 30 f0 68  4a
1a11 : 29 1f 4c 7d d5 0a 10 ef  5a
1a19 : 68 2c a9 5e 85 6b 8a 48  3c
1a21 : 98 48 a5 ae 20 60 d6 a9  d0
1a29 : 00 85 af a5 6b 0a 0a 0a  d0
1a31 : 85 6a a5 6b 4a 4a 4a 4a  77
1a39 : 4a 18 65 fb 85 6b a9 00  c2
1a41 : a8 85 5f 85 5d 85 5e a5  fb
1a49 : 93 0a 0a 2a 85 61 6a 85  c1
1a51 : 5c 20 3f c7 a5 93 85 9e  d0
1a59 : a9 80 85 6d e6 5e d0 02  fa
1a61 : e6 5f a5 5c 18 65 ae 85  5e
1a69 : 5c a5 5d 69 00 85 5d d0  60
1a71 : 06 a5 61 c5 5c b0 0e a5  2a
1a79 : 5c 38 e5 61 85 5c b0 02  99
1a81 : c6 5d 20 47 d6 a5 92 85  d7
1a89 : 57 a5 6d 25 fc f0 20 a5  d6
1a91 : 15 c9 01 90 08 d0 18 a5  90
1a99 : 14 c9 40 b0 12 a5 11 d0  ec
1aa1 : 0e a5 02 aa c9 c8 b0 07  0b
1aa9 : 98 48 20 4f d0 68 a8 20  8b
1ab1 : 40 d6 c6 57 d0 d3 46 6d  98
1ab9 : 90 cb a5 92 0a 0a 0a 20  44
1ac1 : 6a d6 20 50 d6 c6 9e d0  68
1ac9 : 8f c8 c0 08 90 83 a5 93  d0
1ad1 : 0a 0a 0a 20 80 d6 68 a8  19
1ad9 : 68 aa a5 9c 4c 60 d6 e6  84
1ae1 : 14 d0 02 e6 15 60 a5 14  ce
1ae9 : d0 02 c6 15 c6 14 60 e6  6b
1af1 : 02 d0 02 e6 11 60 a5 02  67
1af9 : d0 02 c6 11 c6 02 60 18  cc
1b01 : 65 14 85 14 90 02 e6 15  33
1b09 : 60 49 ff 38 65 14 85 14  4a
1b11 : b0 02 c6 15 60 18 65 02  77
1b19 : 85 02 90 02 e6 11 60 49  0f
1b21 : ff 38 65 02 85 02 b0 02  05
```

```
1b29 : c6 11 60 a9 00 85 60 85  7e
1b31 : 61 85 62 85 63 85 64 a9  e5
1b39 : 80 85 fb 20 ad d6 a9 00  b7
1b41 : 85 fb 85 64 20 ad d6 a9  d0
1b49 : 00 85 68 60 38 26 64 b0  da
1b51 : fa 24 fb 30 03 a5 5c 2c  89
1b59 : a5 5d 25 64 f0 ef a5 5e  65
1b61 : 85 66 a5 5f 85 67 a9 00  a9
1b69 : 85 68 66 67 66 66 a5 64  a2
1b71 : 26 66 26 67 26 68 6a 90  b1
1b79 : f7 18 24 fb 30 1b a5 60  38
1b81 : 65 66 85 60 a5 61 65 67  50
1b89 : 85 61 a5 62 65 68 85 62  e9
1b91 : a5 63 69 00 85 63 4c ae  44
1b99 : d6 a5 61 65 66 85 61 a5  aa
1ba1 : 62 65 67 85 62 a5 63 65  ec
1ba9 : 68 85 63 4c ae d6 a9 00  7e
1bb1 : 85 6c 85 6d 85 0a a5 5b  71
1bb9 : 85 6e a5 6c 85 69 a5 6d  81
1bc1 : 85 6a a5 6e 85 6b a5 0a  11
1bc9 : 20 5b d8 a5 5a 20 6c d8  8c
1bd1 : 85 6c a9 00 65 63 85 6d  59
1bd9 : a9 14 38 e5 0a 20 5b d8  18
1be1 : a5 5b 20 6c d8 85 6e 20  7d
1be9 : d5 d7 20 1c d8 20 f1 d7  3b
1bf1 : 20 31 d8 20 73 d2 20 e3  fa
1bf9 : d7 20 1c d8 20 01 d8 20  b0
1c01 : 31 d8 20 73 d2 20 d5 d7  4a
1c09 : 20 11 d8 20 f1 d7 20 27  99
1c11 : d8 20 73 d2 20 e3 d7 20  f1
1c19 : 11 d8 20 01 d8 20 27 d8  9b
1c21 : 20 73 d2 a4 0a c8 84 0a  51
1c29 : c0 15 90 8e a5 92 85 14  97
1c31 : a5 93 85 15 a5 ae 38 e5  20
1c39 : 5b 85 02 aa 20 4f d0 a5  38
1c41 : ae 18 65 5b 85 02 aa 20  13
1c49 : 4f d0 a5 ae 85 02 aa 18  83
1c51 : a5 92 65 5a 85 14 a5 93  9b
1c59 : 69 00 85 15 20 4f d0 a6  d3
1c61 : 02 a5 92 38 e5 5a 85 14  51
1c69 : a5 93 e9 00 85 15 20 4f  72
1c71 : d0 4c e7 c7 a5 92 18 65  74
1c79 : 69 85 14 a5 93 65 6a 85  78
1c81 : 15 60 a5 92 38 e5 69 85  e5
1c89 : 14 a5 93 e5 6a 85 15 60  f9
1c91 : a5 92 18 65 6c 8d ee dd  dd
1c99 : a5 93 65 6d 8d ef dd 60  9f
1ca1 : a5 92 38 e5 6c 8d ee dd  05
1ca9 : a5 93 e5 6d 8d ef dd 60  cf
1cb1 : a5 ae 18 65 6b 85 02 aa  a0
1cb9 : 4c 4f d0 a5 ae 38 e5 6b  b1
1cc1 : 85 02 aa 4c 4f d0 a5 ae  eb
1cc9 : 18 65 6e 8d f0 dd aa 60  4a
1cd1 : a5 ae 38 e5 6e 8d f0 dd  6b
1cd9 : aa 60 a6 fb 86 02 60 a5  22
1ce1 : a4 85 14 a5 a5 85 15 60  9d
1ce9 : a6 02 e4 fb 60 a5 15 c5  5c
1cf1 : a5 90 06 d0 04 a5 14 c5  c3
1cf9 : a4 60 0a aa bd c4 dd 85  2a
1d01 : 5c bd c5 dd 85 5d a9 00  53
1d09 : 85 5f 60 85 5e 20 8c d6  cd
1d11 : 26 61 26 62 26 63 18 a9  ef
1d19 : 00 24 61 10 01 38 65 62  b2
1d21 : 85 6e 60 20 2a da d0 f4  a0
1d29 : 84 09 a9 40 85 66 a9 90  f8
1d31 : 85 69 a9 e0 85 5c a0 bf  2e
1d39 : 20 ff d9 20 39 da a6 63  9f
1d41 : e0 c7 f0 0f e8 86 02 20  2e
1d49 : 2a da d0 07 a6 02 86 63  51
1d51 : 4c a1 d8 20 7a da 84 a4  82
1d59 : 20 92 da 84 a5 20 92 da  65
1d61 : a5 a5 d0 04 98 d0 58 2c  57
1d69 : 84 a5 20 7a da a5 a4 d0  26
1d71 : 06 98 f0 05 4c 78 d9 84  99
1d79 : a4 20 57 da 20 6a da d0  c1
1d81 : 05 c6 63 4c be d8 a5 a5  e0
1d89 : d0 17 a5 a4 f0 05 20 47  29
1d91 : da d0 a8 20 4e da 20 d2  e3
1d99 : d9 20 47 da 20 47 da d0  f9
1da1 : 9a a5 a4 d0 06 20 4e da  a1
1da9 : 4c 9c d8 20 16 da 20 ba  ac
1db1 : d9 20 39 da 20 42 da 20  04
1db9 : 2a da d0 ef 4c 9c d8 a5  db
1dc1 : 63 85 92 a5 a4 d0 28 20  f2
1dc9 : 7a da f0 12 20 4e da e6  dc
1dd1 : 63 20 d2 d9 20 47 da c6  69
1dd9 : 63 20 92 da d0 18 a5 92  d6
1de1 : 85 63 20 d2 d9 20 47 da  ec
1de9 : e6 63 4c 9c d8 c6 63 20  b9
1df1 : 7a da d0 e5 f0 e8 20 6a  75
1df9 : da f0 f2 a5 63 85 93 a5  b9
1e01 : 92 85 63 20 57 da a5 63  dc
1e09 : c5 93 f0 05 c6 63 4c 64  f6
1e11 : d9 a5 92 85 63 d0 ce a5  55
1e19 : 63 85 92 a5 a5 d0 0f a5  00
1e21 : 92 85 63 20 d2 d9 20 4e  6c
1e29 : da 4c 49 d9 c6 63 20 92  e4
1e31 : da f0 ec 20 7a da f0 e7  d4
1e39 : 20 6a da f0 ef a5 63 85  28
```

Listing zum Programm Grafik 2000 (Fortsetzung)

```
1e41 : 93 a5 92 85 63 20 57 da  46
1e49 : a5 63 c5 93 f0 05 c6 63  9d
1e51 : 4c a6 d9 a5 92 85 63 d0  a0
1e59 : cd a4 6b b1 66 85 63 b1  0d
1e61 : 69 85 61 a4 5e b1 5c a0  a0
1e69 : 00 25 0a f0 01 c8 84 62  ca
1e71 : 60 a4 6b a5 63 91 66 a5  5a
1e79 : 61 91 69 a4 5e b1 5c 85  82
1e81 : 02 a5 0a a6 62 f0 04 05  75
1e89 : 02 d0 04 49 ff 25 02 91  72
1e91 : 5c a5 6b 18 69 01 85 6b  29
1e99 : c9 50 d0 0c a0 ff a9 00  f1
1ea1 : 85 6b 84 67 84 6a 84 5d  52
1ea9 : 4a 4a 4a 85 5e a5 6b 20  5d
1eb1 : 36 d0 85 0a 60 a5 6b 38  43
1eb9 : e9 01 85 6b b0 ea a9 4e  97
1ec1 : a0 bf c4 67 d0 da 68 68  b5
1ec9 : 60 20 11 d0 85 08 a0 00  b3
1ed1 : b1 fd 25 08 f0 01 c8 60  c6
1ed9 : a5 61 85 14 a5 62 85 15  c0
1ee1 : 60 a6 63 86 02 60 e6 61  bf
1ee9 : d0 02 e6 62 60 a5 61 d0  1b
1ef1 : 02 c6 62 c6 61 60 20 39  d4
1ef9 : da 20 42 da 20 11 d0 a0  de
1f01 : 00 11 fd 91 fd 60 4c 6b  26
1f09 : d0 a6 63 f0 09 ca 86 02  28
1f11 : 20 39 da 4c 2a da a0 01  0c
1f19 : 60 a6 62 d0 04 a4 61 f0  4c
1f21 : f5 a4 61 d0 01 ca 88 86  70
1f29 : 15 84 14 20 42 da 4c 2a  0a
1f31 : da a4 61 a6 62 f0 04 c0  ca
1f39 : 3f f0 db c8 d0 e9 e8 d0  a2
1f41 : e6 e4 9c 90 34 a5 92 85  98
1f49 : 14 a5 93 85 15 20 4e d8  03
1f51 : f0 02 b0 25 24 09 70 05  69
1f59 : 10 06 20 41 db 20 30 db  d4
1f61 : 20 41 db 20 3b d8 20 30  78
1f69 : db 20 41 db 20 57 d6 20  78
1f71 : 50 d6 20 49 d8 f0 05 90  a8
1f79 : 03 4c e7 c7 20 30 db 20  c8
1f81 : 41 db 20 40 d8 20 30 db  c7
1f89 : 20 41 db 20 1b db 85 0a  ff
1f91 : 20 40 d6 20 47 d6 20 4e  d3
1f99 : d8 f0 13 20 47 d6 20 1b  94
1fa1 : db 85 09 20 40 d6 a6 02  df
1fa9 : 20 4f d0 4c f4 da a5 0a  ff
1fb1 : 85 09 a6 02 20 4f d0 4c  fd
1fb9 : d0 da a6 02 20 f2 cf 85  c4
1fc1 : 08 a0 00 b1 fd 25 08 f0  5b
1fc9 : 03 a9 00 2c a9 80 60 a6  94
1fd1 : 92 a4 93 a5 a4 85 92 a5  5b
1fd9 : a5 85 93 86 a4 84 a5 60  bc
1fe1 : a6 9c a5 fb 85 9c 86 fb  0d
1fe9 : 60 4c e7 c7 38 a5 92 e5  29
1ff1 : a4 aa a5 93 e5 a5 a8 90  15
1ff9 : f0 18 8a 65 61 85 57 aa  3a
2001 : 98 65 62 85 58 c9 01 90  8e
2009 : 06 d0 de e0 40 b0 da 38  b0
2011 : a5 fb e5 9c 90 d3 18 65  94
2019 : 63 85 59 b0 cc c9 c8 b0  4b
2021 : c8 a5 62 c5 a5 90 31 d0  52
2029 : 06 a5 61 c5 a4 90 29 20  c7
2031 : 30 db a6 61 a4 62 a5 57  c7
2039 : 85 61 a5 58 85 62 86 57  17
2041 : 84 58 a9 55 8d 01 dc a9  ae
2049 : 5c 8d 07 dc a9 47 8d 04  dc
2051 : dc a9 4e 8d 0a dc d0 14  3a
2059 : a9 55 8d 07 dc a9 5c 8d  99
2061 : 01 dc a9 47 8d 0a dc a9  14
2069 : 4e 8d 04 dc a5 63 c5 9c  e0
2071 : 90 12 a9 e6 8d 2c dc 8d  1a
2079 : 2e dc a9 c6 8d 30 dc 8d  41
2081 : 32 dc d0 1b 20 41 db a6  82
2089 : 63 a5 59 85 63 86 59 a9  e9
2091 : e6 8d 30 dc 8d 32 dc a9  17
2099 : c6 8d 2c dc 8d 2e dc 20  ca
20a1 : 55 dc 20 47 da 20 5c dc  2f
20a9 : 20 4e da a5 92 c5 a4 d0  e7
20b1 : 06 a5 93 c5 a5 f0 09 20  6e
20b9 : 24 dc 4c 06 dc 20 24 dc  38
20c1 : 4c e7 c7 a5 59 85 63 a5  42
20c9 : fb 85 5a e6 5a e6 63 c6  f2
20d1 : 5a c6 63 a5 5a c5 9c f0  44
20d9 : 06 20 40 dc 4c 30 dc 20  95
20e1 : 40 d8 a6 5a 86 02 20 1b  b1
20e9 : db 85 09 20 39 da 20 42  3d
20f1 : da 4c 4f d0 e6 a4 d0 02  ba
20f9 : e6 a5 60 a5 a4 d0 02 c6  e5
2101 : a5 c6 a4 60 e0 01 f0 08  28
2109 : a2 a0 a0 e0 a9 fe d0 06  21
2111 : a2 e0 a0 a0 a9 be 8c ee  00
2119 : c7 8e 17 dd 8c 1b dd 8d  dd
2121 : 7a c9 8d 7f c9 8c 2d d0  2a
2129 : 8c aa d1 8c d5 d1 8c d1  d2
2131 : c7 4c e7 c7 86 02 e0 b4  76
2139 : b0 f7 a5 15 c9 01 90 08  e8
2141 : d0 ef a5 14 c9 29 b0 e9  71
2149 : a5 fd 85 69 a5 fe 85 6a  b9
2151 : a2 00 86 09 86 fc 20 d0  29
2159 : dc 20 d0 dc 20 d0 dc a9  64
2161 : 18 20 6a d6 e6 02 a6 fc  12
2169 : e8 e0 15 90 e7 b0 c2 a2  6d
2171 : 00 86 fb 20 1b db 49 ff  6d
2179 : 0a a4 09 b1 69 2a 91 69  4f
2181 : 20 40 d6 a6 fb e8 e0 08  e6
2189 : 90 e7 e6 09 60 20 0b dd  d7
2191 : 86 09 b1 fd aa b1 14 91  73
2199 : fd 8a 91 14 c8 d0 f3 e6  73
21a1 : 15 e6 fe c6 09 d0 eb 4c  21
21a9 : e7 c7 20 b4 c7 a2 20 a0  66
21b1 : 00 84 fd 84 14 a9 a0 85  1f
21b9 : 15 a9 e0 85 fe 60 bd 49  08
21c1 : dd 8d 2e dd 8d 40 dd 20  3f
21c9 : 0b dd ca b1 14 11 fd 91  90
21d1 : fd c8 d0 f7 e6 15 e6 fe  16
21d9 : ca d0 f0 a0 40 b1 14 11  5f
21e1 : fd 91 fd 88 10 f7 30 bf  38
21e9 : 11 31 51 00 00 40 01 80  ee
21f1 : 02 c0 03 00 05 40 06 80  80
21f9 : 07 c0 08 00 0a 40 0b 80  32
2201 : 0c c0 0d 00 0f 40 10 80  e5
2209 : 11 c0 12 00 14 40 15 80  98
2211 : 16 c0 17 00 19 40 1a 80  4a
2219 : 1b c0 1c 00 1e 80 40 20  c3
2221 : 10 08 04 02 01 f8 7c 3e  bd
2229 : 1f 0f 07 03 01 80 c0 e0  cb
2231 : f0 00 00 28 00 50 00 78  9a
2239 : 00 a0 00 c8 00 f0 00 18  5a
2241 : 01 40 01 68 01 90 01 b8  ba
2249 : 01 e0 01 08 02 30 02 58  56
2251 : 02 80 02 a8 02 d0 02 f8  ca
2259 : 02 20 03 48 03 70 03 98  26
2261 : 03 c0 03 00 00 0b 0a 06  12
2269 : 14 e2 1d 8e 27 fc 30 1c  5b
2271 : 3a e1 42 3d 4b 21 53 82  e4
2279 : 5a 55 61 8e 67 23 6d 0d  07
2281 : 72 42 76 bc 79 77 7c 6d  6a
2289 : 7e 9b 7f 00 80 00 00 00  bd
2291 : 00 00 00 00 00 00 00 ff  91
```

Listing zum Programm Grafik 2000 (Schluß). Bei der Eingabe bitte den MSE verwenden.

Die 80-Zeichenkarte zum Abtippen

Verbreitern Sie den Bildschirm Ihres Monitors um das Doppelte: 80 Zeichen pro Zeile. Egal ob Sie nun Programme eingeben oder laufen lassen.

Scheuen Sie sich nicht, das etwas längere MSE-Programm abzutippen. Die Mühe wird mit einer professionellen Bildschirmdarstellung belohnt: 80 Zeichen, nicht nur bei der Ausgabe, sondern auch zur Eingabe. Natürlich entspricht die Bildqualität dieses Programms nicht der einer »richtigen« 80-Zeichenkarte auf Hardware-Basis, man kann aber dennoch gut damit arbeiten.

Die Software-80-Zeichenkarte wurde so im Speicher des C 64 untergebracht, daß kein Basic-Speicherplatz verlorengeht. Das Programm benutzt im einzelnen folgende Speicherbereiche:

$A000-$A7FF : Zeichensatz
$C000-$CFFF : Color-RAM
$D800-$DFFF : Video-RAM
$02A8-$02A9 : Kommunikationsadressen

Haben Sie die 80-Zeichenkarte mit dem MSE abgetippt und gespeichert, können Sie das Programm mit LOAD "80-ZEICHEN",8 laden und mit RUN starten. Das Programm verschiebt sich dann in die vorgesehenen Speicherbereiche und startet.

Mit dieser Routine können Sie Programme genauso bearbeiten wie mit dem normalen — 40spaltigen — Bildschirmeditor. Einige Kleinigkeiten, die im folgenden näher erläutert werden sollen, sind aber zu beachten. So ist es zum Beispiel nicht mehr möglich, mittels der Tastenkombination SHIFT und CBM zwischen Groß- und Kleinschrift umzuschalten. Man muß dazu auf die bekannten Steuercodes 14 und 142 ausweichen. PRINT CHR$(14) schaltet Klein-/Großschreibung ein, PRINT CHR$(142) entsprechend den Groß-/Grafikmodus. Den Klein-/Großschriftmodus kann man aber auch über die Tastenkombination CTRL-N anwählen. CTRL-N entspricht dem ASCII-Code 14. Für die Zurückschaltung gibt es leider keine Tastenkombination.

Auch das Anwählen der Hintergrundfarbe erfolgt etwas ungewohnt. Und zwar nicht über die bekannte Adresse 53281, sondern mit POKE 683,Farbcode. Gibt man diesen Befehl ein, so geschieht zunächst einmal nichts. Erst durch die Eingabe eines zusätzlichen Steuercodes (wiederum wahlweise mittels PRINT-Anweisung oder über eine Control-Tastenkombination) wird der Hintergrund entweder ganz oder teilweise eingefärbt. Gibt man CTRL-G ein, so nimmt der Bildschirmhintergrund die in Adresse 683 festgelegte Farbe an. Drückt man hingegen CTRL-P (für Paint), so wird nur die Fläche eingefärbt, die entweder vom Cursor überschrieben wird, oder durch Scrolling von unten ins Bildfeld tritt. Damit ist es möglich, auch verschiedene Hintergrundfarben darzustellen.

Um auch mit Programmen, die einen 40-Zeichen-Bildschirm verwenden, arbeiten zu können, ist es möglich, wahlweise zwischen den beiden Formaten hin und her zu schalten. Dies ist zum einen mit den Tastenkombinationen CTRL-B (um auf die normale 40-Zeichen-Darstellung zu

schalten) und mit CTRL-A (Umschaltung von 40 auf 80 Zeichen je Zeile) möglich, wobei jedoch einfach nur zwischen den jeweiligen Bildschirmen hin und her geschaltet wird. Damit ist es dann aber nicht möglich, den 40spaltigen Bildschirm zu bearbeiten. Man kann auch mittels »SYS 685« die 80-Zeichenkarte zeitweise abschalten, um mit dem normalen Editor zu arbeiten (der Inhalt des Bildschirmspeichers bleibt übrigens, während die 80-Zeichenkarte aktiv ist, erhalten). Die analoge Rückschaltung erfolgt mittels »SYS 680«.

Um die 80-Zeichendarstellung zu inaktivieren und um alle verbogenen Zeiger wieder zu restaurieren, gibt man »SYS 49155« ein. Diesen Befehl sollte man immer einem STOP-RESTORE vorziehen, denn sonst stürzt der Computer ab. Der Neustart erfolgt dann mit »SYS 49152«.

Der Geräteadresse 6 (die zweite Zahl beim OPEN-Befehl) wurden Steueraufgaben zugewiesen. Dies ist dann nötig, wenn man im 40-Zeichenmodus Kommandos an das aktive Programm (nach der Umschaltung mit »SYS 685« bleibt das Programm aktiv) sendet. Dies ist vor allem dann recht praktisch, wenn man mit zwei Bildschirmen arbeitet. In dem folgenden Beispiel wird vom normalen Editor aus, auf die 80-Zeichen-Darstellung umgeschaltet, dessen Bildschirm gelöscht und wieder auf die 40 Zeichen geschaltet:

OPEN 6,6 : PRINT#6, CHR$(1);CHR$(147);CHR(2) : CLOSE 6

Abschließend möchte ich noch zwei Steuerzeichen anführen, die übrigens alle wahlweise mit dem CHR$-Befehl oder über die Control-Taste eingegeben werden können. Dies sind CHR$(8) und CHR$(9) — beziehungsweise CTRL-I und CTRL-H —, mit denen der Cursor ein- und ausgeschaltet werden kann. Dies ist besonders bei Eingaben mit GET sehr nützlich, da hier ja normalerweise kein Cursor zu sehen ist. Setzt man vor solch einer Eingabeschleife ein »PRINT CHR$(9)« (Cursor on), so erscheint der Zeiger auch während der Eingabe. Danach sollte man ihn mit CTRL-H (oder »PRINT CHR$(8)«) wieder abschalten, da er sonst die Geschwindigkeit beim Listen verlangsamt.

(Matthias Eichholz/Christoph Sauer/hm)

CTRL	CHR$	Wirkung
a	1	80-Zeichen-Darstellung
b	2	40-Zeichen-Darstellung
f	6	Moduswechsel
g	7	Gesamthintergrund anpassen
h	8	Kein Cursor bei GET sichtbar
i	9	Cursor immer vorhanden
n	14	Kleinschreibung
p	16	Farbe nur unter Cursor anpassen
n	142	Großschreibung

Befehlsübersicht der 80-Zeichen-Softkarte

```
programm : 80 zeichen          0801 1981
----------------------------------------
0801 : 0e 08 c0 07 9e 28 32 31  7b
0809 : 31 35 29 3a 00 00 00 00  67
0811 : 00 00 00 00 00 00 00 00  12
0819 : 00 00 00 00 00 00 00 00  1a
0821 : 00 00 00 00 00 00 00 00  22
0829 : 00 00 00 00 00 00 00 00  2a
0831 : 00 00 00 00 00 00 00 00  32
0839 : 00 00 00 00 00 00 00 00  3a
0841 : 00 00 a9 00 85 61 a9 a0  f7
0849 : 85 62 a9 84 85 14 a9 08  aa
0851 : 85 15 a2 08 a0 00 b1 14  03
0859 : 91 61 c8 d0 f9 e6 15 e6  e0
0861 : 62 ca d0 f2 a2 09 a9 c0  55
0869 : 85 62 b1 14 91 61 c8 d0  f7
0871 : f9 e6 15 e6 62 ca d0 f2  a5
0879 : 20 00 c0 20 44 a6 18 4c  40
0881 : 4b a8 00 44 aa ee ee 88  98
0889 : aa 44 00 04 04 4a 2e 6a  f6
0891 : aa 6a 00 0c 8a 8a cc aa  77
0899 : aa cc 00 04 0a 48 a8 88  c1
08a1 : aa 44 00 0c 2a 2a 6a aa  e2
08a9 : aa 6c 00 0e 08 48 ac e8  93
08b1 : 88 6e 00 0e 68 48 ec 48  3f
08b9 : 48 48 00 04 0a 68 ae 6a  19
08c1 : 2a c4 00 8a 8a ca ae aa  ae
08c9 : aa aa 00 0e 44 04 44 44  88
08d1 : 44 6e 00 02 22 02 22 22  8c
08d9 : 2a a4 40 8a 8a ac ac cc  11
08e1 : aa aa 00 08 88 88 88 88  e1
08e9 : 88 ce 00 0a 0e ae ee ea  02
08f1 : aa aa 00 0a 0a ce ae ae  61
08f9 : aa aa 00 0e 0a 4a aa aa  ad
0901 : aa 4e 00 0c 0a ca ac a8  4f
0909 : c8 88 80 04 0a 6a aa aa  aa
0911 : 6e 26 20 0c 0a 6a 8c 8a  57
0919 : 8a 8a 00 06 08 68 84 42  04
0921 : 22 cc 00 4e 44 e4 44 44  78
0929 : 44 64 00 0a 0a aa aa aa  d7
0931 : aa 6e 00 0a 0a aa aa aa  4a
0939 : ee 44 00 0a 0a aa aa ee  09
0941 : ee aa 00 0a 0a aa a4 4a  e3
0949 : aa aa 00 0a 0a aa ae a4  84
0951 : 64 44 80 0e 02 e2 24 48  12
0959 : 88 ee 00 ee 88 88 88 88  36
0961 : 88 ee 00 44 aa 88 cc 88  1c
0969 : aa ee 00 ee 22 22 22 22  68
0971 : 22 ee 00 44 ee 44 44 44  3e
0979 : 44 44 00 00 00 44 ff 44  8a
0981 : 00 00 00 00 00 00 00 00  82
0989 : 00 00 00 44 44 44 44 00  89
0991 : 44 44 00 aa aa aa 00 00  4d
0999 : 00 00 00 44 44 ee 44 ee  cd
09a1 : 44 44 00 44 66 cc ee 66  e5
09a9 : cc 44 00 00 aa 22 44 88  75
09b1 : aa 00 00 44 aa aa 66 bb  f5
09b9 : bb ee 00 22 44 88 00 00  b8
09c1 : 00 00 00 22 44 88 88 88  c1
09c9 : 44 22 00 88 44 22 22 22  52
09d1 : 44 88 00 00 aa 44 ee 44  6b
09d9 : aa 00 00 00 44 44 ee 44  2e
09e1 : 44 00 00 00 00 00 00 00  26
09e9 : 22 22 44 00 00 00 ee 00  e9
09f1 : 00 00 00 00 00 00 00 00  f2
09f9 : 66 66 00 22 22 44 44 44  b5
0a01 : 88 88 00 44 aa aa aa aa  56
0a09 : aa 44 00 22 66 ee 22 22  c4
0a11 : 22 22 00 44 aa 22 44 88  ab
0a19 : 88 ee 00 44 aa 22 44 22  b2
0a21 : aa 44 00 88 aa aa ee 22  fe
0a29 : 22 22 00 ee 88 88 cc 22  7e
0a31 : aa 44 00 22 44 88 cc aa  53
0a39 : aa 44 00 ee 22 22 44 44  b0
0a41 : 44 44 00 44 aa aa 44 aa  96
0a49 : aa 44 00 44 aa aa 66 22  7c
0a51 : aa 44 00 00 [illegible] [illegible] 00 00  51
0a59 : 66 00 00 00 00 22 00 00  d1
0a61 : 22 22 44 00 22 44 88 88  1d
0a69 : 44 22 00 00 00 ee 00 ee  14
0a71 : 00 00 00 00 88 44 22 22  e9
0a79 : 44 88 00 44 aa 22 44 44  df
0a81 : 00 44 00 00 00 00 00 ff  a3
0a89 : 00 00 00 40 44 ae ee a4  50
0a91 : a4 ae 00 c4 a4 a4 c4 a4  f1
0a99 : a4 c4 04 40 a0 80 8f 80  f6
0aa1 : a0 40 00 c0 a0 af a0 a0  c5
0aa9 : a0 c0 00 e0 8f 80 c0 80  c6
0ab1 : 80 e0 00 e0 80 80 c0 80  ce
0ab9 : 80 8f 00 48 a8 88 e8 a8  ce
0ac1 : a8 48 08 a1 a1 a1 e1 a1  b5
0ac9 : a1 a1 01 e0 40 40 40 43  25
0ad1 : 44 e8 08 22 22 22 22 21  ce
0ad9 : a0 40 00 a1 a1 c1 c2 cc  9a
0ae1 : a0 a0 00 88 88 88 88 88  e2
0ae9 : 88 e8 0f a8 e8 e4 e4 a2  4d
0af1 : a2 a1 01 a1 a1 e2 e2 e4  5f
0af9 : a4 a8 08 ef a8 a8 a8 a8  b5
0b01 : a8 e8 08 cf a1 a1 c1 81  4a
0b09 : 81 81 01 46 a6 af af af  5a
0b11 : ef 66 06 c0 a0 a0 c0 a0  20
0b19 : a0 af 00 60 8a 8e 4e 2e  50
0b21 : 24 c4 00 e8 48 48 48 48  3d
0b29 : 48 48 08 a0 a0 a0 a0 a0  7e
0b31 : a1 e2 02 a9 a9 a6 a6 a6  b1
0b39 : e6 49 09 a6 a6 a9 a9 e9  0d
0b41 : e9 a6 06 a0 a4 aa 4a a4  25
0b49 : a4 ae 00 a2 a2 a2 e2 42  e8
0b51 : 42 42 02 e4 24 2e 4e 8e  dc
0b59 : 84 e4 00 22 22 22 22 ff  4f
0b61 : 22 22 22 44 88 44 88 44  fb
0b69 : 88 44 88 22 22 22 22 22  7a
0b71 : 22 22 22 50 50 a0 ae 5a  b0
0b79 : 5a aa a0 9f 4f 27 97 43  57
0b81 : 23 91 41 00 00 00 00 00  bd
0b89 : 00 00 00 cc cc cc cc cc  23
0b91 : cc cc cc 00 00 00 00 00  f7
0b99 : 00 ff ff ff 00 00 00 00  99
0ba1 : 00 00 00 00 00 00 00 00  a2
0ba9 : 00 00 ff 88 88 88 88 88  ba
0bb1 : 88 88 88 aa 55 aa 55 aa  4a
0bb9 : 55 aa 55 11 11 11 11 11  db
0bc1 : 11 11 11 00 00 00 00 aa  f5
0bc9 : 55 aa 55 9f 2f 4e 9e 2c  f5
0bd1 : 4c 98 28 11 11 11 11 11  96
0bd9 : 11 11 11 22 22 22 22 33  1e
0be1 : 22 22 22 00 00 00 00 33  03
0be9 : 33 33 33 22 22 22 33 00  c7
0bf1 : 00 00 00 00 00 00 00 cc  8b
0bf9 : 44 44 44 00 00 00 00 00  71
0c01 : 00 ff ff 00 00 00 00 33  67
0c09 : 22 22 22 22 22 22 22 ff  c5
0c11 : 00 00 00 00 00 00 00 ff  11
0c19 : 22 22 22 22 22 22 22 ee  b3
0c21 : 22 22 22 88 88 88 88 88  ee
0c29 : 88 88 88 cc cc cc cc cc  b1
0c31 : cc cc cc 33 33 33 33 33  fd
0c39 : 33 33 33 ff ff 00 00 00  d3
0c41 : 00 00 00 ff ff ff 00 00  41
0c49 : 00 00 00 00 00 00 00 00  4a
0c51 : ff ff ff 01 01 21 21 a1  51
0c59 : a1 c1 cf 00 00 00 00 cc  68
0c61 : cc cc cc 33 33 33 33 00  c6
0c69 : 00 00 00 22 22 22 22 ee  47
0c71 : 00 00 00 cc cc cc cc 00  71
0c79 : 00 00 00 cc cc cc cc 33  e0
0c81 : 33 33 33 bb 55 11 11 77  a3
0c89 : 55 bb ff fb fb b5 d1 95  1b
0c91 : 55 95 ff f3 75 75 33 55  aa
0c99 : 55 33 ff fb f5 b7 57 77  70
0ca1 : 55 bb ff f3 d5 d5 95 55  5f
0ca9 : 55 93 ff f1 f7 b7 53 17  bf
0cb1 : 77 91 ff f1 97 b7 13 b7  22
0cb9 : b7 b7 ff fb f5 97 51 95  58
0cc1 : d5 3b ff 75 75 35 51 55  d3
0cc9 : 55 55 ff f1 bb fb bb bb  09
0cd1 : bb 91 ff fd dd fd dd dd  15
0cd9 : d5 5b bf 75 75 53 53 33  a0
0ce1 : 55 55 ff f7 77 77 77 77  e0
0ce9 : 77 31 ff f5 f1 51 11 15  cf
0cf1 : 55 55 ff f5 f5 31 51 51  80
0cf9 : 55 55 ff f1 f5 b5 55 55  44
0d01 : 55 b1 ff f3 f5 35 53 57  b2
0d09 : 37 77 7f fb f5 95 55 55  67
0d11 : 91 d9 df f3 f5 95 73 75  ca
0d19 : 75 75 ff f9 f7 97 7b bd  2d
0d21 : dd 33 ff b1 bb 1b bb bb  c9
0d29 : bb 9b ff f5 f5 55 55 55  7a
0d31 : 55 91 ff f5 f5 55 55 55  17
0d39 : 11 bb ff f5 f5 55 55 11  68
0d41 : 11 55 ff f5 f5 55 5b b5  9e
0d49 : 55 55 ff f5 f5 55 51 5b  0d
0d51 : 9b bb 7f f1 fd 1d db b7  8f
0d59 : 77 11 ff 11 77 77 77 77  7b
0d61 : 77 11 ff bb 55 77 33 77  a5
0d69 : 55 11 ff 11 dd dd dd dd  69
0d71 : dd 11 ff bb 11 bb bb bb  a3
0d79 : bb bb ff ff ff bb 00 bb  67
0d81 : ff ff ff ff ff ff ff ff  80
0d89 : ff ff ff bb bb bb bb ff  88
0d91 : bb bb ff 55 55 55 ff ff  d4
0d99 : ff ff ff bb bb 11 bb 11  64
0da1 : bb bb ff bb 99 33 11 99  5c
0da9 : 33 bb ff ff 55 dd bb 77  dc
0db1 : 55 ff ff bb 55 55 99 44  6c
0db9 : 44 11 ff dd bb 77 ff ff  b9
0dc1 : ff ff ff dd bb 77 77 77  c0
```

Listing der 80-Zeichen-Softwarekarte. Beachten Sie bitte bei der Eingabe den MSE.

```
0dc9 : bb dd ff 77 bb dd dd dd  3f
0dd1 : bb 77 ff ff 55 bb 11 bb  37
0dd9 : 55 ff ff ff bb bb 11 bb  83
0de1 : bb ff ff ff ff ff ff ff  9c
0de9 : dd dd bb ff ff ff 11 ff  e8
0df1 : ff ff ff ff ff ff ff ff  f0
0df9 : 99 99 ff dd dd bb bb bb  3c
0e01 : 77 77 ff bb 55 55 55 55  ab
0e09 : 55 bb ff dd 99 11 dd dd  4d
0e11 : dd dd ff bb 55 dd bb 77  76
0e19 : 77 11 ff bb 55 dd bb dd  7f
0e21 : 55 bb ff 77 55 55 11 dd  43
0e29 : dd dd ff 11 77 77 33 dd  d3
0e31 : 55 bb ff dd bb 77 33 55  0e
0e39 : 55 bb ff 11 dd dd bb bb  c1
0e41 : bb bb ff bb 55 55 bb 55  eb
0e49 : 55 bb ff bb 55 55 99 dd  15
0e51 : 55 bb ff ff ff 99 ff ff  50
0e59 : 99 ff ff ff ff dd ff ff  e0
0e61 : dd dd bb ff dd bb 77 77  a4
0e69 : bb dd ff ff ff 11 ff 11  bd
0e71 : ff ff ff ff 77 bb dd dd  f8
0e79 : bb 77 ff bb 55 dd bb bb  12
0e81 : ff bb ff ff ff ff ff 00  5e
0e89 : ff ff ff bf bb 51 11 5b  c1
0e91 : 5b 51 ff 3b 5b 5b 3b 5b  30
0e99 : 5b 3b fb bf 5f 7f 70 7f  3b
0ea1 : 5f bf ff 3f 5f 50 5f 5f  7c
0ea9 : 5f 3f ff 1f 70 7f 3f 7f  8b
0eb1 : 7f 1f ff 1f 7f 7f 3f 7f  94
0eb9 : 7f 70 ff b7 57 77 17 57  a3
0ec1 : 57 b7 f7 5e 5e 5e 1e 5e  cc
0ec9 : 5e 5e fe 1f bf bf bf bc  6c
0ed1 : bb 17 f7 dd dd dd dd de  d3
0ed9 : 5f bf ff 5e 5e 3e 3d 33  17
0ee1 : 5f 5f ff 77 77 77 77 77  df
0ee9 : 77 17 f0 57 17 1b 1b 5d  84
0ef1 : 5d 5e fe 5e 5e 1d 1d 1b  82
0ef9 : 5b 57 f7 10 57 57 57 57  3c
0f01 : 57 17 f7 30 5e 5e 3e 7e  b7
0f09 : 7e 7e fe b9 59 50 50 50  b7
0f11 : 10 99 f9 3f 5f 5f 3f 5f  01
0f19 : 5f 50 ff 9f 75 71 b1 d1  e1
0f21 : db 3b ff 17 b7 b7 b7 b7  04
0f29 : b7 b7 f7 5f 5f 5f 5f 5f  d3
0f31 : 5e 1d fd 56 56 59 59 59  b0
0f39 : 19 b6 f6 59 59 56 56 16  64
0f41 : 16 59 f9 5f 5b 55 b5 5b  5c
0f49 : 5b 51 ff 5d 5d 5d 1d bd  a9
0f51 : bd bd fd 1b db d1 b1 71  c5
0f59 : 7b 1b ff dd dd dd dd 00  62
0f61 : dd dd dd bb 77 bb 77 bb  c6
0f69 : 77 bb 77 dd dd dd dd dd  57
0f71 : dd dd dd af af 5f 51 a5  31
0f79 : a5 55 5f 60 b0 d8 68 bc  9a
0f81 : dc 6e be ff ff ff ff ff  44
0f89 : ff ff ff 33 33 33 33 33  ee
0f91 : 33 33 33 ff ff ff ff ff  2b
0f99 : ff 00 00 00 ff ff ff ff  98
0fa1 : ff ff ff ff ff ff ff ff  a0
0fa9 : ff ff 00 77 77 77 77 77  97
0fb1 : 77 77 77 55 aa 55 aa 55  17
0fb9 : aa 55 aa ee ee ee ee ee  96
0fc1 : ee ee ee ff ff ff ff 55  8c
0fc9 : aa 55 aa 60 d0 b1 61 d3  9c
0fd1 : b3 67 d7 ee ee ee ee ee  0b
0fd9 : ee ee ee dd dd dd dd cc  93
0fe1 : dd dd dd ff ff ff ff cc  be
0fe9 : cc cc cc dd dd dd cc ff  0a
0ff1 : ff ff ff ff ff ff ff 33  56
0ff9 : bb bb bb ff ff ff ff ff  81
1001 : ff 00 00 ff ff ff ff cc  9a
1009 : dd dd dd dd dd dd dd 00  4c
1011 : ff ff ff ff ff ff ff 00  10
1019 : dd dd dd dd dd dd dd 11  7e
1021 : dd dd dd 77 77 77 77 77  53
1029 : 77 77 77 33 33 33 33 33  a0
1031 : 33 33 33 cc cc cc cc cc  64
1039 : cc cc cc 00 00 ff ff ff  9e
1041 : ff ff ff 00 00 00 ff ff  40
1049 : ff ff ff ff ff ff ff ff  48
1051 : 00 00 00 fe fe de de 5e  50
1059 : 5e 3e 30 ff ff ff ff 33  49
1061 : 33 33 33 cc cc cc cc ff  fb
1069 : ff ff ff dd dd dd dd 11  8a
1071 : ff ff ff 33 33 33 33 ff  70
1079 : ff ff ff 33 33 33 33 cc  11
1081 : cc cc cc 4c 06 c0 4c 15  32
1089 : c6 a9 42 8d 26 03 a9 c0  09
1091 : 8d 27 03 a9 bb 8d 24 03  67
1099 : a9 c7 8d 25 03 a9 01 8d  ca
10a1 : ac 02 ad 00 c0 8d a8 02  d9
10a9 : 8d ad 02 a9 fd 8d a9 02  ba
10b1 : a9 c5 8d aa 02 a9 f5 8d  56
10b9 : ae 02 a9 c5 8d af 02 20  2a
10c1 : fd c0 4c 8e c8 85 02 a5  0f
10c9 : 9a c9 06 f0 0e c9 03 d0  c4
10d1 : 05 ad ac 02 d0 05 a5 02  e8
10d9 : 4c ca f1 8a 48 98 48 a5  0e
10e1 : 02 20 cd c6 20 6f c0 20  00
10e9 : 9d c6 68 a8 68 aa a5 02  8f
10f1 : 18 60 c9 20 90 53 c9 40  fb
```

```
10f9 : 90 36 c9 60 90 19 c9 80  1d
1101 : 90 1a c9 a0 90 43 c9 c0  f0
1109 : 90 17 c9 e0 90 18 c9 ff  a4
1111 : d0 19 a9 5e 4c 9f c2 e9  45
1119 : 3f 4c 9f c2 e9 1f 4c 9f  c6
1121 : c2 e9 3f 4c 9f c2 e9 7f  e8
1129 : 4c 9f c2 e9 7f 4c 9f c2  91
1131 : c9 22 f0 03 4c 9f c2 aa  ca
1139 : ad 74 c1 d0 08 ad 73 c1  ea
1141 : 49 01 8d 73 c1 8a 4c 9f  bd
1149 : c2 4c 27 c4 00 00 40 01  97
1151 : 80 02 c0 03 00 05 40 06  98
1159 : 80 07 c0 08 00 0a 40 0b  f5
1161 : 80 0c c0 0d 00 0f 40 10  53
1169 : 80 11 c0 12 00 14 40 15  b0
1171 : 80 16 c0 17 00 19 40 1a  0d
1179 : 80 1b c0 1c 00 1e 40 1f  6b
1181 : ea ad 11 d0 8d 62 c1 ad  ee
1189 : 18 d0 8d 63 c1 ad 16 d0  5d
1191 : 8d 64 c1 ad 00 dd 8d 65  66
1199 : c1 29 fc 8d 00 dd a9 3b  ec
11a1 : 8d 11 d0 a9 48 8d 18 d0  13
11a9 : a9 60 8d fd c0 a9 08 8d  3a
11b1 : 16 d0 60 ad 62 c1 8d 11  8a
11b9 : d0 ad 63 c1 8d 18 d0 ad  a9
11c1 : 64 c1 8d 16 d0 ad 65 c1  bf
11c9 : 8d 00 dd a9 ea 8d fd c0  97
11d1 : 60 a5 01 8d 66 c1 29 fc  09
11d9 : 09 04 78 85 01 60 ad 66  4a
11e1 : c1 85 01 58 60 1b 15 c8  75
11e9 : 03 37 10 00 00 00 00 e0  4e
11f1 : 00 d0 00 d8 00 00 00 00  75
11f9 : 00 00 00 00 00 00 00 00  fa
1201 : d8 00 4f 00 00 00 01 00  b1
1209 : 80 a9 e0 8d 6c c1 a9 d0  65
1211 : 8d 6e c1 a9 d8 8d 70 c1  ba
1219 : a9 00 8d 69 c1 8d 6a c1  09
1221 : 8d 68 c1 8d 6b c1 8d 6d  da
1229 : c1 8d 6f c1 60 20 4e c1  89
1231 : 20 86 c1 a8 20 e7 c1 98  93
1239 : 91 fc c8 d0 fb e6 fd d0  25
1241 : f7 20 f2 c1 a2 04 ad 67  0d
1249 : c1 91 fc c8 d0 fb e6 fd  af
1251 : ca d0 f6 20 2c c6 20 fd  ba
1259 : c1 a2 08 a9 20 91 fc c8  b7
1261 : d0 fb e6 fd ca d0 f6 4c  50
1269 : 5b c1 ad 6b c1 85 fc ad  15
1271 : 6c c1 85 fd 60 ad 6d c1  8b
1279 : 85 fc ad 6e c1 85 fd 60  b6
1281 : ad 6f c1 85 fc ad 70 c1  89
1289 : 85 fd 60 ad 6b c1 8d [illegible]  b0
1291 : c1 8d 6f c1 ad 6c c1 38  e3
1299 : e9 e0 4a 6e 6d c1 4a 6e  3e
12a1 : 6d c1 4a 6e 6d c1 48 18  85
12a9 : 69 d0 8d 6e c1 ad 6d c1  6e
12b1 : 8d 6f c1 68 0e 6f c1 2a  2b
12b9 : aa ad 6f c1 18 6d 68 c1  60
12c1 : 8d 6f c1 8a 69 d8 8d 70  3c
12c9 : c1 60 ee 6a c1 ae 68 c1  7a
12d1 : e8 8a 29 01 8d 68 c1 d0  2e
12d9 : 18 ad 6b c1 18 69 08 8d  e3
12e1 : 6b c1 ad 6c c1 69 00 8d  a8
12e9 : 6c c1 ad 6a c1 c9 50 b0  fb
12f1 : 03 4c 08 c2 a9 00 8d 6a  1a
12f9 : c1 8d 68 c1 ae 69 c1 e8  e2
1301 : e0 19 90 06 20 0c c3 ae  22
1309 : 69 c1 8e 69 c1 8a 0a aa  11
1311 : bd c9 c0 8d 6b c1 bd ca  e6
1319 : c0 18 69 e0 8d 6c c1 4c  38
1321 : 08 c2 ae 74 c1 ca 30 03  fe
1329 : 8e 74 c1 ae 71 c1 f0 02  24
1331 : 09 80 48 20 4e c1 20 e7  d4
1339 : c1 68 85 fe 48 a9 00 06  4d
1341 : fe 2a 06 fe 2a 06 fe 2a  d9
1349 : 18 69 a0 85 ff a0 07 b1  73
1351 : fe 2c 72 c1 10 04 0a 0a  97
1359 : 0a 0a 29 f0 85 fb b1 fc  ca
1361 : ae 68 c1 d0 04 29 0f 10  b4
1369 : 0a 29 f0 46 fb 46 fb 46  7b
1371 : fb 46 fb 05 fb 91 fc 88  80
1379 : 10 d5 c8 20 f2 c1 ad 67  6d
1381 : c1 91 fc 20 fd c1 68 91  01
1389 : fc 20 5b c1 4c 47 c2 a0  f0
1391 : e0 84 fd c8 84 ff ac 7a  3c
1399 : c1 f0 01 60 a0 00 84 fc  35
13a1 : a9 40 85 fe 20 4e c1 a2  6c
13a9 : 1e b1 fe 91 fc c8 d0 f9  df
13b1 : e6 fd e6 ff ca d0 f2 20  8f
13b9 : 2c c6 98 99 00 fe c8 d0  5e
13c1 : fa a0 40 99 00 ff 88 10  91
13c9 : fa 20 2c c6 c8 a9 cf 85  db
13d1 : fd 85 ff a9 c0 85 fc a9  45
13d9 : e8 85 fe a2 04 a0 40 b1  41
13e1 : fe 91 fc c8 d0 f9 e6 fd  74
13e9 : e6 ff ca d0 f2 ad 67 c1  59
13f1 : a2 27 9d c0 d3 ca 10 fa  70
13f9 : a9 d8 85 fd 85 ff 84 fc  94
1401 : a9 50 85 fe a2 08 b1 fe  43
1409 : 91 fc c8 d0 f9 e6 fd e6  01
1411 : ff ca d0 f2 a9 20 a2 4f  cc
1419 : 9d 80 df ca 10 fa 20 5b  58
1421 : c1 4c a1 c3 ad 8e 02 d0  e2
```

```
1429 : fb 60 ae 69 c1 d0 01 60  94
1431 : ce 69 c1 38 ad 6b c1 e9  3c
1439 : 40 8d 6b c1 ad 6c c1 e9  6c
1441 : 01 8d 6c c1 4c 08 c2 ad  c8
1449 : 6a c1 f0 0d ce 6a c1 ce  56
1451 : 68 c1 f0 33 a9 01 8d 68  e6
1459 : c1 ae 69 c1 d0 06 ad 6a  cd
1461 : c1 d0 01 60 38 ad 6b c1  f9
1469 : e9 08 8d 6b c1 ad 6c c1  e6
1471 : e9 00 8d 6c c1 ad 6a c1  02
1479 : d0 0d ce 69 c1 a9 4f 8d  72
1481 : 6a c1 29 01 8d 68 c1 4c  f2
1489 : 08 c2 ae 69 c1 e0 18 b0  b0
1491 : 16 ad 6b c1 69 40 8d 6b  36
1499 : c1 ad 6c c1 69 01 8d 6c  32
14a1 : c1 ee 69 c1 4c 08 c2 4c  15
14a9 : 0c c3 c9 0d f0 34 c9 8d  9e
14b1 : f0 42 ae 73 c1 d0 0c ae  0c
14b9 : 74 c1 f0 18 c9 94 d0 0a  e6
14c1 : 4c 4b c5 c9 14 d0 03 4c  ca
14c9 : bd c5 18 69 80 90 02 69  03
14d1 : bf 4c 9f c2 c9 01 d0 03  e4
14d9 : 4c fd c0 c9 02 d0 03 4c  d8
14e1 : 30 c1 c9 0d d0 0e a9 00  2a
14e9 : 8d 73 c1 8d 74 c1 8d 71  c0
14f1 : c1 4c 71 c2 c9 8d d0 0e  f5
14f9 : a9 00 8d 73 c1 8d 74 c1  52
1501 : 8d 71 c1 4c 71 c2 c9 0e  b1
1509 : d0 06 a9 00 8d 72 c1 60  7b
1511 : c9 12 d0 06 a9 01 8d 71  94
1519 : c1 60 c9 92 d0 06 a9 00  b3
1521 : 8d 71 c1 60 c9 11 d0 03  52
1529 : 4c 07 c4 c9 91 d0 03 4c  a7
1531 : a7 c3 c9 1d d0 03 4c 47  b5
1539 : c2 c9 9d d0 03 4c c4 c3  8e
1541 : c9 13 d0 03 4c 86 c1 c9  bc
1549 : 93 d0 03 4c aa c1 c9 8e  8c
1551 : d0 06 a9 80 8d 72 c1 60  d3
1559 : c9 94 d0 03 4c 4b c5 c9  cb
1561 : 14 d0 03 4c bd c5 c9 08  69
1569 : d0 05 a9 80 8d 85 c1 c9  d6
1571 : 09 d0 05 a9 00 8d 85 c1  5f
1579 : c9 07 d0 03 4c 79 c6 c9  9a
1581 : 10 d0 03 4c 65 c6 c9 06  03
1589 : d0 03 4c 05 c6 4c 38 c6  cc
1591 : ad 6b c1 8d 77 c1 ad 6c  2b
1599 : c1 8d 78 c1 ad 69 c1 8d  bf
15a1 : 75 c1 ad 6a c1 8d 76 c1  95
15a9 : ad 68 c1 8d 79 c1 60 ad  2f
15b1 : 77 c1 8d 6b c1 ad 78 c1  c8
15b9 : 8d 6c c1 ad 75 c1 8d 69  11
15c1 : c1 ad 76 c1 8d 6a c1 ad  bd
15c9 : 79 c1 8d 68 c1 60 ee 7a  63
15d1 : c1 20 0d c5 a9 00 8d 68  40
15d9 : c1 ad 69 c1 0a aa bd c9  84
15e1 : c0 8d 6b c1 bd ca c0 18  e0
15e9 : 69 e0 8d 6c c1 20 08 c2  76
15f1 : 20 4e c1 20 fd c1 a0 4f  bc
15f9 : b1 fc c9 20 d0 36 88 b1  e3
1601 : fc c8 91 fc 88 cc 76 c1  b1
1609 : d0 f4 20 5b c1 20 2c c5  20
1611 : 20 08 c2 a9 20 20 af c2  62
1619 : a0 00 20 fd c1 20 4e c1  5b
1621 : a0 00 b1 fc 48 20 5b c1  44
1629 : 68 20 af c2 ad 6a c1 d0  bc
1631 : e9 ee 74 c1 ce 7a c1 20  ef
1639 : 5b c1 20 2c c5 4c 08 c2  67
1641 : ad 6a c1 f0 30 20 c4 c3  50
1649 : 20 0d c5 ee 7a c1 20 fd  71
1651 : c1 20 4e c1 a0 01 b1 fc  c1
1659 : 48 20 5b c1 68 20 af c2  8c
1661 : ac 6a c1 c0 4f d0 e7 a9  39
1669 : 20 20 af c2 ce 7a c1 20  e5
1671 : 2c c5 4c 08 c2 4c c4 c3  bd
1679 : a9 00 8d ac 02 4c 30 c1  e2
1681 : a9 01 8d ac 02 4c fd c0  a0
1689 : ad ac 02 49 01 8d ac 02  69
1691 : f0 03 4c fd c0 4c 30 c1  88
1699 : a9 ca 8d 26 03 a9 f1 8d  30
16a1 : 27 03 a9 57 8d 24 03 a9  f8
16a9 : f1 8d 25 03 4c f5 c5 a9  e9
16b1 : 37 8d fa ff a9 c6 8d fb  6c
16b9 : ff 60 40 a0 0f d9 55 c6  af
16c1 : f0 04 88 10 f8 60 98 0a  e0
16c9 : 0a 0a 0a 85 fb ad 67 c1  5a
16d1 : 29 0f 05 fb 8d 67 c1 60  1e
16d9 : 90 05 1c 9f 9c 1e 1f 9e  5b
16e1 : 81 95 96 97 98 99 9a 9b  bd
16e9 : ad ab 02 29 0f 8d ab 02  22
16f1 : ad 67 c1 29 f0 0d ab 02  11
16f9 : 8d 67 c1 60 a2 04 a9 d0  49
1701 : 85 fd a0 00 84 fc 20 4e  fa
1709 : c1 20 65 c6 b1 fc 29 f0  96
1711 : 0d ab 02 91 fc c8 d0 f4  ea
1719 : e6 fd ca d0 ef 4c 5b c1  1d
```

Listing der 80-Zeichen-Softwarekarte (Fortsetzung). Beachten Sie bitte bei der Eingabe den MSE.

```
1721 : 2c 85 c1 30 2a 48 20 4e  88
1729 : c1 20 fd c1 a0 00 b1 fc  7d
1731 : 48 8d 7b c1 20 f2 c1 b1  5b
1739 : fc 8d 83 c1 20 5b c1 68  ca
1741 : 49 80 ee 7a c1 20 af c2  37
1749 : 20 c4 c3 ce 7a c1 68 60  ae
1751 : 2c 85 c1 30 fa 48 ee 7a  59
1759 : c1 ad 7b c1 20 af c2 20  d3
1761 : c4 c3 20 4e c1 20 f2 c1  45
1769 : ad 83 c1 a0 00 91 fc 20  1d
1771 : 5b c1 ce 7a c1 68 60 ad  ec
1779 : 69 c1 0a aa bd c9 c0 8d  e3
1781 : 7c c1 bd ca c0 4a 6e 7c  b8
1789 : c1 4a 6e 7c c1 18 69 d8  cf
1791 : 8d 7d c1 60 20 f4 c6 ad  79
1799 : 69 c1 8d 80 c1 ad 6a c1  0d
17a1 : 8d 81 c1 2c 85 c1 10 03  91
17a9 : 20 a2 c6 a5 c6 f0 fc 2c  c1
17b1 : 85 c1 10 03 20 d2 c6 20  6f
17b9 : b4 e5 c9 0d f0 1b c9 06  8f
17c1 : f0 e1 85 02 20 58 c0 ad  67
17c9 : 69 c1 cd 80 c1 d0 3c ad  85
17d1 : 6a c1 cd 81 c1 90 34 b0  92
17d9 : ca 20 d2 c6 a9 00 8d 68  e2
17e1 : c1 8d 6a c1 ae 69 c1 20  b9
17e9 : 87 c2 a0 4f 20 4e c1 20  9f
17f1 : b0 c7 b1 fc c9 20 d0 03  78
17f9 : 88 10 f7 c8 8c 7f c1 a0  ad
```

```
1801 : 01 8c 7e c1 88 8c 82 c1  9b
1809 : 4c 5b c1 20 f4 c6 a9 00  a3
1811 : 8d 81 c1 ad 69 c1 8d 80  61
1819 : c1 4c 20 c7 85 d7 29 3f  3c
1821 : 06 d7 24 d7 10 02 09 80  4d
1829 : 90 04 a6 d4 d0 04 70 02  f2
1831 : 09 40 60 ad 7c c1 85 fc  0e
1839 : ad 7d c1 85 fd 60 ad ac  b9
1841 : 02 f0 08 a5 99 f0 07 c9  43
1849 : 03 f0 03 4c 57 f1 8a 48  ce
1851 : 98 48 20 dc c7 85 02 68  32
1859 : a8 68 aa a5 02 18 60 a5  42
1861 : 99 c9 03 f0 07 ad 7e c1  19
1869 : f0 16 d0 1a 20 f4 c6 ac  fa
1871 : 6a c1 8c 81 c1 20 67 c7  59
1879 : a0 00 8c 82 c1 4c 03 c8  a9
1881 : 20 fd c0 20 11 c7 a6 d4  68
1889 : ad 82 c1 8e 82 c1 85 d4  af
1891 : 20 4e c1 20 b0 c7 ac 81  4c
1899 : c1 b1 fc 48 20 5b c1 20  9f
18a1 : 47 c2 68 c9 5e f0 1b 20  b7
18a9 : 99 c7 48 20 84 e6 ee 81  7a
18b1 : c1 cc 7f c1 f0 10 a5 d4  c0
18b9 : ae 82 c1 86 d4 8d 82 c1  31
18c1 : 68 60 a9 ff d0 e4 c8 8c  34
18c9 : 6a c1 98 48 a9 00 8d 7e  11
18d1 : c1 85 02 ad 69 c1 0a aa  ad
18d9 : 68 48 29 fe 0a 26 02 0a  7d
```

```
18e1 : 26 02 7d c9 c0 8d 6b c1  4a
18e9 : bd ca c0 65 02 69 e0 8d  f2
18f1 : 6c c1 68 29 01 8d 68 c1  1f
18f9 : 20 08 c2 a4 99 c0 03 d0  b0
1901 : 03 20 71 c2 28 20 a2 c6  65
1909 : 20 d2 c6 a9 0d 48 4c 33  24
1911 : c8 ea a9 a4 a0 c8 20 1e  5a
1919 : ab a9 00 8d ab 02 8d 20  8c
1921 : d0 a9 60 8d 8e c8 60 93  67
1929 : 0e 08 9b 20 20 20 20 20  ea
1931 : 20 2a 2a 2a 2a 2a 2a 20  13
1939 : 20 d5 2d d0 52 20 38 30  11
1941 : 2d c3 4f 4c 55 4d 4e 20  e6
1949 : d3 43 52 45 45 4e 20 28  93
1951 : 20 c4 45 56 23 20 30 2f  42
1959 : 33 2f 36 20 29 20 42 59  05
1961 : 20 d5 52 41 4e 55 53 20  46
1969 : d0 52 4f 47 52 41 4d 53  2a
1971 : 20 2a 2a 2a 2a 2a 2a 11  35
1979 : 11 00 00 00 00 00 00 00  8b
```

Listing der 80-Zeichen-Softwarekarte (Schluß). Beachten Sie bitte bei der Eingabe den MSE.

Charakter-Designer für den C 64

Mit diesem Programm passen Sie komfortabel alle darstellbaren Bildschirmzeichen Ihren eigenen Vorstellungen an.

Der Charakter-Designer besteht aus drei Programmteilen. Einem Basic-Lader zum Laden des Hauptprogramms, dem Hauptprogramm selbst, das unbedingt unter dem Namen CHARAKTER-MAIN abgespeichert werden muß und einem Zeichensatz-Lader. Mit ihm können selbstdefinierte Zeichensätze außerhalb des Charakter-Designers geladen werden. Dazu muß der Name des Zeichensatzes eingegeben werden. Ist der Ladevorgang beendet, wird gefragt, ob die Zeichen im Multicolor-Modus ausgegeben werden sollen oder nicht. Wird der Multicolor-Modus gewählt, können nur die Farben, die über die Commodore-Taste erreichbar sind (8 bis 15) genutzt werden. Dadurch lassen sich Multicolor- und normale Zeichen gleichzeitig darstellen.

Der Charakter-Designer ist so aufgebaut, daß während des Programmablaufs eine Bildschirmmaske bestehen bleibt.

Der Bildschirm besteht aus folgenden Feldern:

— dem Sketch-Pad:
  Zeichenfeld, in dem vier Zeichen auf einmal gezeichnet werden können;
— dem Farbfeld:
  Anzeige der ausgewählten Farben und der aktuellen Farbe, mit der gezeichnet wird;
— einer Anzeige der vier in Arbeit stehenden Zeichen in Originalgröße;
— einer Anzeige der vier in Arbeit stehenden Zeichen in doppelter Größe;
— dem Charakterwahlfeld:
  Anzeige des gesamten Zeichensatzes;
— der Statuszeile:
  Ein- und Ausgabezeile für Disketteninhalt, Eingabe der Namen für LOAD und SAVE und zur Ausgabe von Fehlermeldungen;
— einer Anzeige der am häufigsten gebrauchten Funktionen.

Die Steuerung des Cursorpfeils erfolgt mit dem Joystick (Port 2) oder mit den Cursortasten (Punkt setzen mit Space).

Funktionen:

Zur Beschreibung der Funktionen einige Erklärungen im voraus:

Der »aktuelle Charakter« ist das Zeichen, über dem sich der Cursorpfeil auf dem Sketch-Pad gerade befindet.

Im »Normalmodus« besteht ein Zeichen aus 64 Grafikpunkten (8x8) und kann eine von 16 Farben annehmen. Im »Multicolormodus« wird die halbe Auflösung des Charakters genommen (8x4 Punkte). In diesem Modus können vier Farben auf einmal in einem Zeichen verwendet werden, wobei die vierte Farbe der des Hintergrunds entspricht. Leider kann bei der Farbe »1« nur unter acht Farben ausgewählt werden.

M (Multi-/Normalcolorumschaltung)
Durch diese Funktion kann zwischen Normal- und Multicolormodus gewählt werden.

D (Drehen)
Durch diese Funktion ist es möglich, ein Zeichen um 90 Grad nach links zu drehen (nur im Normalmodus).

SHIFT-D (Drehen)
Wie bei D, jedoch wird das ganze Sketch-Pad gedreht.

F (Spiegeln)
Der aktuelle Charakter erscheint spiegelverkehrt auf dem Bildschirm.

SHIFT-F (Spiegeln)
Wie bei F, jedoch wird das ganze Sketch-Pad an der Y-Achse gespiegelt.

K (Kopfstellen)
Der aktuelle Charakter wird an der X-Achse gespiegelt.

SHIFT-K (Kopfstellen)
Wie bei F, jedoch wird das ganze Sketch-Pad an der X-Achse gespiegelt.

G (Groß/Klein)
Mit dieser Taste kann zwischen Groß- und Kleinschrift selektiert werden. Wenn der abgespeicherte Charaktersatz später, außerhalb des Programms, geladen wird, ist der Zeichensatz aktiv, der zuletzt im Designer bearbeitet wurde und kann mit SHIFT-Commodore umgeschaltet werden.

I (Invers)
Diese Funktion bewirkt, daß der aktuelle Charakter invertiert wird. Im Multicolormodus werden die Farben auch entsprechend geändert.

SHIFT-I
Wie bei I, jedoch wird das ganze Sketch-Pad invers gesetzt.

Commodore-Taste (Charakter auswählen)
Durch Drücken der Commodore-Taste kann ein neues Zeichen in das aktuelle Feld genommen werden. Der Pfeil springt in die obere linke Ecke des Charakter-Auswahlfeldes. Nun kann man mit dem Pfeil einen Charakter auswählen, der mit Return übernommen wird.

C (Copy) — Kopieren eines Zeichens
Im Copy-Modus springt der Pfeil in die linke Ecke des Charakterauswahlfeldes. Nun kann das Zeichen ausgewählt werden, das man kopieren will. Nach dem Drücken der RETURN-Taste kann man entscheiden, wo das Zeichen hinkopiert werden soll.

L (Laden eines Zeichensatzes)
Es erscheint »Name:« in der Statuszeile. Nun kann man den Filenamen über die Tastatur eingeben. Nach Drücken der RETURN-Taste (Eingabe beendet) schaltet der Bildschirm ab und der Zeichensatz wird geladen. Falls ein Diskettenfehler auftritt, erscheint die Meldung »Diskerror« in der Statuszeile. Durch Drücken der SPACE-Taste kann im Programm fortgefahren werden (die ganze Fehlermeldung des Floppy-Laufwerks kann nicht ausgegeben werden, weil die Statuszeile dafür zu klein ist).

S (Save)
Wie bei L, jedoch wird der Zeichensatz abgelegt.

***** (Directory lesen)
Mit dieser Funktion kann das Inhaltsverzeichnis angezeigt werden. Die einzelnen Namen werden in der Statuszeile angezeigt, wobei der erste Ausdruck der Diskettenname ist. Durch Drücken von SPACE (Leertaste) wird der nächste Filename angezeigt.

@ (Initialisieren)
Der Original-C 64-Zeichensatz wird aus dem ROM in das RAM übertragen. So können auch kleine Änderungen ohne große Komplikationen durchgeführt werden.

Z (Zurück)
Mit dieser Funktion kann der zuletzt geladene Zeichensatz zurückgeholt werden.

T (Testen)
Das Auswahlfeld wird gelöscht und der Pfeil verschwindet. Nun kann man mit dem Zeichensatz schreiben, um zu testen, wie er wirkt. Es können alle Tasten benutzt werden. Nach Drücken der RETURN-Taste erscheint der Zeichensatz wieder und man kann im Programm fortfahren.

Q (Quit)
Diese Taste muß (aus Sicherheitsgründen) dreimal gedrückt werden, um das Programm zu beenden. Nach erstmaligem Drücken der Taste erscheint der Ausdruck »ACHTUNG RESET« in der Statuszeile. Wenn man eine andere Taste drückt, verschwindet der Ausdruck wieder und man kann weiterzeichnen.

CLR/HOME (Löschen eines Zeichens)
Die CLR/HOME-Taste hat drei Funktionen:
1. Alleiniges Drücken der Taste bewirkt, daß der Pfeil in die obere linke Ecke des Sketch-Pads springt.
2. Zusammen mit der SHIFT-Taste wird der aktuelle Charakter gelöscht.
3. Zusammen mit der CTRL-Taste wird das gesamte Sketch-Pad gelöscht.

Einfügen und Herausnehmen von Zeilen und Spalten:
Diese Funktion ist eine Besonderheit des Charakter-Designers. Sie ermöglicht es, Punkte in einer Zeile beziehungsweise Spalte einzufügen und herauszunehmen. Der Pfeil kann an einer beliebigen Stelle des Sketch-Pads positioniert werden.

Mit folgenden Tasten wird gearbeitet.
INS/DEL-Taste: für zeilenweises Herausnehmen. Durch zusätzliches Drücken der SHIFT-Taste wird eine Zeile eingefügt. Es werden bis zu 255 Zeilen gepuffert, die automatisch wieder ausgegeben werden.
Pfund-Taste: wie bei **INS/DEL**, jedoch wird der jeweils darunter liegende Charakter mit bewegt.
=-Taste: für das spaltenweise Herausnehmen von einzelnen Punkten. Mit SHIFT »=« wird ein Punkt eingefügt. Hierbei wird nichts gepuffert, weil es in den meisten Fällen, nur hinderlich wäre.
;-Taste: wie bei =, jedoch werden zwei nebeneinander liegende Charaktere bearbeitet.

Farbgebung:
Die Farben werden mit den Tasten 1 bis 4 ausgewählt und können durch zusätzliches Drücken der SHIFT-Taste verändert werden.
Im Normalmodus kann man durch Drücken der Taste 1 die

```
0 GOTO 10                                   <178>
1 ********************************          <163>
2 *    CHARAKTER-DESIGNER C=64     *        <159>
3 *     *** LADER FUER MAIN ***    *        <167>
4 * V ON MATTHIAS MEYER             *       <045>
5 *                                *        <028>
6 * WINKELWEG 24                   *        <096>
7 * 5505 WASSERLIESCH BEI TRIER    *        <251>

9 ********************************          <171>
10 IF A= 1 THEN SYS 8*4096+9: REM START DE
   S CHARAKTER-DESIGNERS                    <105>
20 A= 1: LOAD "CHARAKTER-MAIN",8,1          <213>

© 64'er
```

Listing 1. Charakter-Design-Lader. Bei der Eingabe bitte den Checksummer anwenden.

```
0 GOTO 10                                   <178>
1 ********************************          <163>
2 *    CHARAKTER-DESIGNER C=64     *        <159>
3 *   *** ZEICHENSATZ LADER ***    *        <208>
4 * V ON  MATTHIAS MEYER           *        <045>
5 *                                *        <028>
6 * WINKELWEG 24                   *        <096>
7 * 5505 WASSERLIESCH  BEI TRIER   *        <251>

9 ********************************          <171>
10 IF A=0 THEN 100: REM SCHON GELADEN ?     <228>
20 POKE 53248+24, 29: REM  CHARAKTERSATZ E
   INSCHALTEN                               <104>
30 PRINT"MULTICOLOR (J/N) ";:POKE 198,0: W
   AIT 198,1:GET MU$: POKE 198,0            <050>
40 IF MU$="J"THEN POKE VIC+22, 216: GOTO 6
   0: REM  CHA. AUF MULTICOLOR SCHALTEN     <221>
50 IF MU$<>"N"THEN PRINT: GOTO 30: REM  BE
   I FALSCHER TASTE ZURUECK                 <076>
60 PRINT MU$: END: REM  ENDE                <127>
70 REM  FUER MULTICOLOR GELTEN NUR DIE FAR
   BEN 8-16 !!                              <047>
80 REM  DIE ANDEREN FARBEN WERDEN IN REG 3
   4 UND 35 FESTGELEGT                      <000>
90 --- LADEN DES ZEICHENSATZES ---          <084>
100 POKE 56, 3*16: POKE 55, 0: CLR: REM  E
    NDE DES BASIK RAM AUF $3000 SETZEN      <059>
110 VIC = 53248:REM  VIC GRUNDADRESSE       <059>
120 POKE VIC+32,9: POKE VIC+33,7: PRINT"{C
    LR,YELLOW}":POKE VIC+33,9: REM  FARBEN
     SETZEN                                 <036>
130 INPUT"CHARAKTERSATZ NAME";NA$: REM  NA
    ME DES CHARAKTERSATZES                  <211>
140 NA$="{HOME}"+NA$: REM  DAS ERKENNUNGSZ
    EICHEN FUER EIN CHA.SATZ VOR DEN NAMEN <080>
150 A = 1:LOAD NA$ ,8,1: REM  CHA.SATZ VON
     FLOPPY LADEN                           <049>

© 64'er
```

Listing 2. Zeichensatzlader. Bei der Eingabe bitte den Checksummer verwenden.

Farbe eins zum Zeichnen nehmen. Wenn man Punkte löschen will, muß man die Taste 4 drücken.

Im Multicolormodus sind die Tasten 1 bis 3 mit den verschiedenen Farben belegt, wobei mit der Taste 4 Punkte gelöscht werden können.

Programmablauf:

Nach dem Starten (durch Reset oder Restore) werden die Initialisierungsroutinen aufgerufen. Danach werden die einzelnen Unterroutinen in den $C000-Bereich verschoben und die Bildschirmmaske einschließlich Farbe übertragen. Nun werden die programminternen Initialisierungsroutinen aufgerufen (Rasterzeilen-Interrupt, Sprites für Charakteranzeige und so weiter) und der aktuelle Zeichensatz wird aktiviert. Anschließend befindet sich das Programm in der Tastatur- und Joystick-Abfrage (Hauptschleife).

(Matthias Meyer/ah)

```
 programm : charakter-main  8000 9a85
--------------------------------------

8000 : 09 80 09 80 c3 c2 cd 38  96
8008 : 30 8e 16 d0 a9 2a a2 85  a0
8010 : a0 00 85 f7 86 f8 84 f9  47
8018 : a9 c0 a2 09 a0 80 85 fa  05
8020 : 8e 18 03 8c 19 03 8c 03  ef
8028 : 03 8e 02 03 a0 05 b9 0f  8b
8030 : fd 99 03 80 88 d0 f7 a0  fb
8038 : 00 b1 f7 91 f9 c8 d0 f9  5e
8040 : e6 f8 e6 fa a5 fa c9 cf  b4
8048 : d0 ef 78 20 a3 fd 20 50  7d
8050 : fd 20 15 fd 20 5b ff 58  f0
8058 : a9 09 8d 20 d0 8d 21 d0  8d
8060 : a9 8e 20 d2 ff a9 08 20  60
8068 : d2 ff a9 01 85 cc a9 3c  a2
8070 : 8d 05 dc a9 1b 8d 11 d0  f0
8078 : a9 00 8d 00 7e 8d 00 7f  d8
8080 : a9 c8 8d c4 02 20 00 c0  2c
8088 : a2 00 a9 00 9d 3c 03 e8  2e
8090 : e0 c4 d0 f8 a2 00 a0 00  d2
8098 : b9 1d 82 9d c0 03 e8 e8  ce
80a0 : e8 c8 c0 08 d0 f2 a0 00  45
80a8 : 98 99 c0 02 c8 c0 04 d0  c1
80b0 : f7 20 34 c7 a9 00 8d 10  ae
80b8 : d0 a9 07 8d 15 d0 a9 2c  a7
80c0 : a2 3d a0 02 8d 00 d0 8e  a2
80c8 : 01 d0 8c 1d d0 8c 17 d0  67
80d0 : a9 b9 a2 42 8d 02 d0 8e  90
80d8 : 03 d0 a9 c0 a2 72 8d 04  c2
80e0 : d0 8e 05 d0 a9 0d a2 0f  fe
80e8 : 8d f9 07 8d fa 07 8e f8  f9
80f0 : 07 ad c4 02 c9 d8 f0 12  8a
80f8 : ad 6c d8 ae 94 d8 ac bc  23
8100 : d8 8d 86 02 8e 22 d0 8c  d7
8108 : 23 d0 20 ef c2 20 7b c3  3c
8110 : a9 e0 8d 02 dc ad 00 dc  c2
8118 : 85 f7 c9 00 f0 16 29 01  72
8120 : f0 2b a5 f7 29 02 f0 35  df
8128 : a5 f7 29 04 f0 3f a5 f7  23
8130 : 29 08 f0 49 a5 f7 29 10  a1
8138 : f0 53 a9 ff 8d 02 dc a2  de
8140 : 1e c8 d0 fd ee 27 d0 ca  b7
8148 : d0 f7 4c 25 82 ad 01 d0  07
8150 : c9 3d f0 d4 38 e9 08 8d  9d
8158 : 01 d0 4c 28 81 ad 01 d0  05
8160 : c9 b5 f0 c4 18 69 08 8d  e1
8168 : 01 d0 4c 28 81 ad 00 d0  11
8170 : c9 2c f0 c0 38 e9 08 8d  b2
8178 : 00 d0 4c 34 81 ad 00 d0  a1
8180 : c9 a4 f0 b0 18 69 08 8d  f6
8188 : 00 d0 4c 34 81 a9 04 a2  45
8190 : 00 85 3c 86 3b ad 00 d0  f5
8198 : 38 e9 18 4a 4a 4a 85 f8  13
81a0 : ad 01 d0 38 e9 32 4a 4a  f7
81a8 : 4a 85 f9 a8 a9 00 18 69  16
81b0 : 28 90 02 e6 3c 85 3b 88  6c
81b8 : d0 f4 18 65 f8 90 02 e6  9f
81c0 : 3c 85 3b 20 ff 81 a0 00  20
81c8 : 91 3b ad c4 02 29 10 f0  86
81d0 : 25 ad 00 d0 c9 a4 f0 06  77
81d8 : c8 20 ff 81 91 3b a5 3c  e2
81e0 : 18 69 d4 85 3c a0 00 ad  b7
81e8 : 0c d9 91 3b ae 00 d0 e0  9d
81f0 : a4 f0 03 c8 91 3b 20 e2  1f
81f8 : c2 20 ef c2 4c 3f 81 ad  3f
8200 : 21 d0 29 0f 8d cc 02 ad  58
8208 : 0c d9 29 0f cd cc 02 f0  5a
8210 : 03 a9 a0 60 a9 20 60 20  79
8218 : ef c2 4c 10 81 fc f8 f0  43
8220 : f8 dc 8e 07 03 a5 cb c9  2b
8228 : 07 d0 1c a5 02 c9 01 b0  27
8230 : 05 e6 02 4c 10 81 a9 00  66
8238 : 85 02 ad 8d 02 c9 01 f0  30
8240 : 03 4c 5d 81 4c 4d 81 c9  ba
8248 : 02 d0 1c a5 02 c9 01 b0  42
8250 : 05 e6 02 4c 10 81 a9 00  86
8258 : 85 02 ad 8d 02 c9 01 f0  50
8260 : 03 4c 7d 81 4c 6d 81 c9  e3
8268 : 3c d0 03 4c 8d 81 ad 8d  0d
8270 : 02 c9 02 d0 20 20 03 c7  90
8278 : a0 00 ad cb 02 d9 c0 02  f3
8280 : d0 10 98 48 20 8f c3 68  e6
8288 : a8 ad cb 02 99 c0 02 4c  7a
8290 : b4 80 c8 d0 e5 a5 cb c5  17
8298 : 92 d0 03 4c 10 81 85 92  25
82a0 : 85 93 c9 24 d0 3c ad c4  15
82a8 : 02 c9 c8 d0 1b a9 d8 8d  58
82b0 : c4 02 ad 6c d8 29 07 8d  7c
82b8 : 6c d8 8d 0c d9 09 08 8d  96
82c0 : 86 02 20 7b c3 4c dc 82  d6
82c8 : a9 c8 8d c4 02 ad 86 02  7d
82d0 : 8d 6c d8 8d 0c d9 8d 86  4e
82d8 : 02 20 7b c3 20 ef c2 4c  67
82e0 : 10 81 a5 cb c9 38 d0 3c  ae
82e8 : ad 8d 02 c9 01 f0 09 ad  2d
82f0 : 6c d8 8d 0c d9 4c 10 81  f0
82f8 : ad c4 02 c9 c8 d0 15 ee  06
8300 : 6c d8 ad 6c d8 8d 0c d9  af
8308 : 8d 86 02 20 7b c3 20 ef  93
8310 : c2 4c 10 81 a9 00 20 8a  5d
8318 : c7 ad 0c d9 29 07 8d 0c  0d
8320 : d9 4c 10 81 c9 3b d0 1f  4c
8328 : ad 8d 02 c9 01 f0 09 ad  6d
8330 : 94 d8 8d 0c d9 4c 10 81  58
8338 : ad c4 02 c9 d8 d0 f6 a9  44
8340 : 01 20 8a c7 4c 10 81 c9  cc
8348 : 08 d0 1f ad 8d 02 c9 01  48
8350 : f0 09 ad bc d8 8d 0c d9  a6
8358 : 4c 10 81 ad c4 02 c9 d8  f8
8360 : d0 d3 a9 02 20 8a c7 4c  d3
8368 : 10 81 c9 0b d0 09 ad 21  5b
8370 : d0 8d 0c d9 4c 10 81 c9  24
8378 : 33 d0 3e ad 8d 02 c9 01  6b
8380 : f0 07 c9 04 b0 0c 4c b4  ed
8388 : 80 20 03 c7 20 78 c7 4c  50
8390 : b4 80 ad c0 02 8d cb 02  c7
8398 : 20 78 c7 ad c1 02 8d cb  96
83a0 : 02 20 78 c7 ad c2 02 8d  dd
83a8 : cb 02 20 78 c7 ad c3 02  88
83b0 : 8d cb 02 20 78 c7 4c b4  08
83b8 : 80 c9 21 d0 14 ad 8d 02  88
83c0 : c9 01 f0 09 20 [illegible] c7 20  e1
83c8 : f8 c4 4c 17 82 20 34 c6  a0
83d0 : 4c 17 82 c9 15 d0 16 ad  0d
83d8 : 8d 02 c9 01 f0 09 20 03  d7
83e0 : c7 20 ae c5 4c 17 82 20  e3
83e8 : 58 c6 4c 17 82 c9 12 d0  fa
83f0 : 1d ad c4 02 c9 d8 f0 f2  62
83f8 : ad 8d 02 c9 01 f0 09 20  22
8400 : 03 c7 20 7f c5 4c 17 82  ff
8408 : 20 c4 c6 4c 17 82 c9 14  9a
8410 : d0 06 20 bb c5 4c b4 80  f5
8418 : c9 25 d0 16 ad 8d 02 c9  4e
8420 : 01 f0 09 20 03 c7 20 54  77
8428 : c5 4c 17 82 20 8e c6 4c  54
8430 : 17 82 c9 2a d0 0d a9 00  5c
8438 : 85 fe 20 36 c8 20 00 c0  1a
8440 : 4c b4 80 c9 0d d0 0a a9  12
8448 : 01 85 fe 20 36 c8 4c b4  14
8450 : 80 c9 2e d0 06 20 b5 c2  18
8458 : 4c 17 82 c9 00 d0 13 ad  38
8460 : 8d 02 c9 01 d0 06 20 06  4b
8468 : ca 4c 17 82 20 0c ca 4c  95
8470 : 17 82 c9 30 d0 13 ad 8d  b8
8478 : 02 c9 01 d0 06 20 cb ca  df
8480 : 4c 17 82 20 fe ca 4c 17  a2
8488 : 82 c9 31 d0 10 20 40 cb  f0
8490 : a9 07 8d 15 d0 a5 ff 8d  18
8498 : 86 02 4c 17 82 c9 36 d0  06
84a0 : 03 4c 58 80 c9 0c d0 24  78
84a8 : a9 00 a2 40 a0 30 85 f7  94
84b0 : 85 f9 86 f8 84 fa a0 00  95
84b8 : b1 f7 91 f9 c8 d0 f9 e6  d1
84c0 : f8 e6 fa a5 fa c9 40 d0  3f
84c8 : ef 4c 17 82 c9 1a d0 06  b0
84d0 : 20 c4 cc 4c 17 82 c9 16  e8
84d8 : d0 06 20 22 cc 4c 17 82  88
84e0 : c9 35 d0 11 ad 8d 02 d0  8b
84e8 : 06 20 7e cd 4c 07 85 20  ab
84f0 : 55 cd 4c 07 85 c9 32 d0  31
84f8 : 24 ad 8d 02 d0 06 20 c6  e2
8500 : cd 4c 07 85 20 a7 cd ac  36
8508 : c4 02 a5 92 c5 93 d0 0a  d9
8510 : a9 00 85 93 a5 92 c0 d8  31
8518 : f0 c6 4c 17 82 c9 3e d0  72
8520 : 06 20 f3 cd 4c 17 82 4c  0d
8528 : 10 81 78 a9 1f a2 c0 8d  71
8530 : 14 03 8e 15 03 a9 c1 8d  ac
8538 : 12 d0 ad 11 d0 29 7f 8d  af
8540 : 11 d0 a9 81 8d 1a d0 58  f1
8548 : 60 ad 19 d0 8d 19 d0 29  16
8550 : 01 d0 07 ad 0d dc 58 4c  e3
8558 : 31 ea ad 12 d0 c9 fb b0  59
8560 : 0d a9 1d 8d 18 d0 a9 fb  e1
8568 : 8d 12 d0 4c 25 c8 a9 15  25
8570 : 8d 18 d0 a9 c1 8d 12 d0  e5
8578 : 4c 2e c8 a9 00 a2 30 c0  9a
8580 : 00 f0 09 18 69 08 90 01  59
8588 : e8 88 d0 f7 85 f7 86 f8  0b
8590 : 60 ac c0 02 20 51 c0 a9  9a
8598 : 29 a2 04 85 f9 86 fa 20  c4
85a0 : 9c c0 ac c2 02 20 51 c0  08
85a8 : 20 9c c0 a9 31 a2 04 85  bf
85b0 : f9 86 fa ac c1 02 20 51  90
85b8 : c0 20 9c c0 ac c3 02 20  f9
85c0 : 51 c0 20 9c c0 60 a2 08  b7
85c8 : a0 08 b1 f9 c9 a0 d0 07  0b
85d0 : 38 6e ca 02 4c b1 c0 18  b8
85d8 : 6e ca 02 88 d0 ec a5 f9  3c
85e0 : 18 69 28 90 02 e6 fa 85  17
85e8 : f9 a0 00 ad ca 02 91 f7  da
85f0 : e6 f7 d0 02 e6 00 ca d0  81
85f8 : cf 60 ac c0 02 20 51 c0  22
8600 : a9 29 a2 04 85 f9 86 fa  9f
8608 : 20 07 c1 ac c2 02 20 51  11
8610 : c0 20 07 c1 a9 31 a2 04  91
8618 : 85 f9 86 fa ac c1 02 20  bc
8620 : 51 c0 20 07 c1 ac c3 02  4f
8628 : 20 51 c0 20 07 c1 4c fc  cf
8630 : c2 a2 08 a0 00 b1 f7 8d  e2
8638 : ca 02 a0 08 6e ca 02 90  93
8640 : 17 a9 a0 91 f9 a5 f9 85  46
8648 : fb a5 fa 18 69 d4 85 fc  25
8650 : ad 86 02 91 fb 4c 32 c1  61
8658 : a9 20 91 f9 88 d0 dd a5  87
8660 : f9 18 69 28 90 02 e6 fa  6f
8668 : 85 f9 e6 f7 d0 02 e6 f8  4d
8670 : ca d0 c0 60 ad 22 d0 29  60
8678 : 0f 8d cb 02 ad 23 d0 29  0a
8680 : 0f 8d cc 02 ac c0 02 20  e2
8688 : 51 c0 a9 29 a2 04 a0 29  e8
8690 : 85 f9 86 fa 84 fb a9 d8  93
8698 : 85 fc 20 9f c1 ac c2 02  28
86a0 : 20 51 c0 20 9f c1 a9 31  ae
86a8 : a2 04 85 f9 86 fa a9 31  36
86b0 : a2 d8 85 fb 86 fc ac c1  26
86b8 : 02 20 51 c0 20 9f c1 ac  96
86c0 : c3 02 20 51 c0 20 9f c1  c6
86c8 : 60 a2 08 a0 08 b1 f9 c9  19
86d0 : a0 d0 2f b1 fb 29 0f cd  bb
86d8 : cb 02 d0 0b 38 6e ca 02  60
86e0 : 18 6e ca 02 4c e0 c1 cd  91
86e8 : cc 02 d0 0b 18 6e ca 02  6f
86f0 : 38 6e ca 02 4c e0 c1 38  95
86f8 : 6e ca 02 38 6e ca 02 4c  31
8700 : e0 c1 18 6e ca 02 18 6e  8f
8708 : ca 02 88 88 d0 bf ad ca  5e
8710 : 02 91 f7 e6 f7 d0 02 e6  91
8718 : f8 a5 f9 18 69 28 90 04  86
8720 : e6 fa e6 fc 85 f9 85 fb  13
8728 : ca d0 a0 60 ad 22 d0 29  10
8730 : 0f 8d cb 02 ad 23 d0 29  c2
8738 : 0f 8d cc 02 ac c0 02 20  9a
8740 : 51 c0 a9 29 a2 04 a0 29  a0
8748 : 85 f9 86 fa 84 fb a9 d8  4b
8750 : 85 fc 20 59 c2 ac c2 02  27
8758 : 20 51 c0 20 59 c2 a9 31  0a
8760 : a2 04 85 f9 86 fa a9 31  ee
8768 : a2 d8 85 fb 86 fc ac c1  de
8770 : 02 20 51 c0 20 59 c2 ac  20
8778 : c3 02 20 51 c0 20 59 c2  67
8780 : 4c fc c2 a2 08 a0 00 b1  38
8788 : f7 8d ca 02 a0 08 4e ca  52
8790 : 02 b0 15 4e ca 02 b0 0a  8d
8798 : a9 20 91 f9 88 91 f9 4c  8a
87a0 : 99 c2 ad cc 02 4c 8c c2  d9
87a8 : 4e ca 02 b0 06 ad cb 02  f3
87b0 : 4c 8c c2 ad 6c d8 91 fb  74
87b8 : 88 91 fb c8 a9 a0 91 f9  fb
87c0 : 88 91 f9 88 d0 c8 a5 f9  7e
87c8 : 18 69 28 90 04 e6 fa e6  e2
```

Listing 3. Charakter-Designer Hauptprogramm. Bei der Eingabe bitte den MSE verwenden.

```
87d0 : fc 85 f9 85 fb e6 f7 d0  36
87d8 : 02 e6 f8 ca d0 a7 60 78  a1
87e0 : a9 31 85 01 a9 00 a2 d0  6a
87e8 : 85 f9 86 fa a9 00 a2 30  f0
87f0 : 85 fb 86 fc a2 10 a0 00  e1
87f8 : b1 f9 91 fb c8 d0 f9 e6  52
8800 : fa e6 fc ca d0 f2 a9 37  bf
8808 : 85 01 58 60 ad c4 02 29  8b
8810 : 10 d0 03 4c 67 c0 4c 4a  15
8818 : c1 ad c4 02 29 10 d0 03  7e
8820 : 4c d0 c0 4c 02 c2 ac c0  f8
8828 : 02 20 51 c0 85 f9 86 fa  df
8830 : ac c1 02 20 51 c0 a0 00  df
8838 : a2 00 b1 f9 9d 40 03 e8  40
8840 : b1 f7 9d 40 03 e8 e8 c8  09
8848 : c0 08 d0 ee ac c2 02 20  47
8850 : 51 c0 85 f9 86 fa ac c3  1c
8858 : 02 20 51 c0 a0 00 a2 18  9b
8860 : b1 f9 9d 40 03 e8 b1 f7  ab
8868 : 9d 40 03 e8 e8 c8 c0 08  eb
8870 : d0 ee ad c4 02 29 10 f0  47
8878 : 1d ad 86 02 29 07 ae 22  18
8880 : d0 ac 23 d0 8d 26 d0 8e  f4
8888 : 25 d0 8c 28 d0 8c 29 d0  f5
8890 : a9 06 8d 1c d0 60 ad 86  f7
8898 : 02 8d 28 d0 8d 29 d0 a9  3e
88a0 : 00 8d 1c d0 60 ad 86 02  1a
88a8 : ea ea aa a0 00 98 99 d0  93
88b0 : 06 8a 99 d0 da c8 d0 f5  9f
88b8 : 60 a9 1c a2 c6 a0 00 8d  d5
88c0 : 00 d0 8e 01 d0 8c cb 02  91
88c8 : 8c 10 d0 a9 e0 8d 02 dc  02
88d0 : ad 00 dc 8d ca 02 29 01  c9
88d8 : d0 06 20 38 c4 4c c0 c3  f4
88e0 : ad ca 02 29 02 d0 03 20  8b
88e8 : 52 c4 ad ca 02 29 04 d0  7c
88f0 : 06 20 b2 c4 4c d7 c3 ad  39
88f8 : ca 02 29 08 d0 03 20 6c  8d
8900 : c4 ad ca 02 29 10 d0 03  ea
8908 : 4c 2e c4 a9 ff 8d 02 dc  00
8910 : a2 32 c8 d0 fd ee 27 d0  ad
8918 : ca d0 f7 a5 cb c9 01 d0  ad
8920 : 03 4c 2e c4 ae 8d 02 e0  8f
8928 : 00 d0 14 c9 07 d0 06 20  1e
8930 : 52 c4 4c 26 c4 c9 02 d0  00
8938 : 03 20 6c c4 4c 26 c4 c9  9c
8940 : 07 d0 06 20 38 c4 4c 26  5c
8948 : c4 c9 02 d0 ef 20 b2 c4  e0
8950 : a9 e0 8d 02 dc 4c a6 c3  5f
8958 : a9 ff a2 00 86 c6 8d 02  82
8960 : dc 60 ad cb 02 c9 28 90  81
8968 : 12 ad cb 02 38 e9 28 8d  13
8970 : cb 02 ad 01 d0 38 e9 08  4e
8978 : 8d 01 d0 60 ad cb 02 c9  9b
8980 : d8 b0 12 ad cb 02 18 69  eb
8988 : 28 8d cb 02 ad 01 d0 18  00
8990 : 69 08 8d 01 d0 60 ad cb  df
8998 : 02 c9 ff f0 19 ee cb 02  d9
89a0 : ad 10 d0 d0 12 ad 00 d0  d4
89a8 : c9 fc d0 26 a9 01 a2 05  1f
89b0 : 8d 10 d0 8e 00 d0 60 ad  af
89b8 : 00 d0 c9 55 d0 14 a9 00  92
89c0 : a2 1c 8d 10 d0 8e 00 d0  f9
89c8 : ad 01 d0 18 69 08 8d 01  3c
89d0 : d0 60 ad 00 d0 18 69 08  bf
89d8 : 8d 00 d0 60 ad cb 02 c9  7a
89e0 : 00 f0 f8 ce cb 02 ad 10  14
89e8 : d0 d0 1b ad 00 d0 c9 1c  83
89f0 : d0 26 a9 01 a2 55 8d 10  89
89f8 : d0 8e 00 d0 ad 01 d0 38  c0
8a00 : e9 08 8d 01 d0 60 ad 00  38
8a08 : d0 c9 05 d0 0b a9 00 a2  5b
8a10 : fc 8d 10 d0 8e 00 d0 60  de
8a18 : ad 00 d0 38 e9 08 8d 00  16
8a20 : d0 60 ac cb 02 20 51 c0  ad
8a28 : a0 00 b1 f7 49 ff 91 f7  fe
8a30 : c8 c0 08 d0 f5 60 ac cb  21
8a38 : 02 20 51 c0 a0 00 b1 f7  77
8a40 : a2 00 0a 6e ca 02 18 e8  22
8a48 : e0 08 d0 f6 ad ca 02 91  9c
8a50 : f7 c8 c0 08 d0 e8 60 ac  0c
8a58 : cb 02 20 51 c0 a0 00 b1  cb
8a60 : f7 a2 00 0a 08 0a 6e ca  0a
8a68 : 02 28 6e ca 02 18 e8 e0  ba
8a70 : 04 d0 f0 ad ca 02 91 f7  c1
8a78 : c8 c0 08 d0 e2 60 ac cb  38
8a80 : 02 20 51 c0 a9 00 a2 07  32
8a88 : 85 f9 86 fa a4 f9 b1 f7  db
8a90 : 48 a4 fa b1 f7 a4 f9 91  cf
8a98 : f7 68 a4 fa 91 f7 e6 f9  b4
8aa0 : c6 fa a5 f9 c9 04 d0 e4  56
8aa8 : 60 ac cb 02 20 51 c0 a0  62
8ab0 : 00 b1 f7 99 a6 c5 c8 c0  f7
8ab8 : 08 d0 f6 a0 00 a2 00 5e  cc
8ac0 : a6 c5 2a e8 e0 08 d0 f7  72
8ac8 : 91 f7 c8 c0 08 d0 ee 60  22
8ad0 : 00 00 00 00 00 00 00 00  d1
8ad8 : ad c4 02 29 10 d0 03 4c  b9
8ae0 : 0c c5 4c 2d c5 20 8f c3  ab
8ae8 : ac cb 02 20 51 c0 ea b1  29
8af0 : f7 99 a6 c5 c8 c0 08 d0  6a
8af8 : f6 a5 cb c9 40 d0 fa a9  b7
```

```
8b00 : e0 8d 02 dc ad 00 dc 29  64
8b08 : 10 f0 f9 20 8f c3 ac cb  74
8b10 : 02 20 51 c0 a0 00 b9 a6  cd
8b18 : c5 91 f7 c8 c0 08 d0 f6  3a
8b20 : 60 8d ca 02 8c cb 02 20  a9
8b28 : 51 c0 a0 00 b1 f7 99 a6  90
8b30 : c5 c8 c0 08 d0 f6 ac ca  97
8b38 : 02 20 51 c0 85 f9 86 fa  ef
8b40 : ac cb 02 20 51 c0 a0 00  f4
8b48 : b1 f9 91 f7 c8 c0 08 d0  ad
8b50 : f7 a0 00 b9 a6 c5 91 f9  a1
8b58 : c8 c0 08 d0 f6 60 ad c0  47
8b60 : 02 8d cb 02 20 f8 c4 ad  94
8b68 : c1 02 8d cb 02 20 f8 c4  96
8b70 : ad c2 02 8d cb 02 20 f8  f0
8b78 : c4 ad c3 02 8d cb 02 4c  1c
8b80 : f8 c4 ad c0 02 8d cb 02  1d
8b88 : 20 0c c5 ad c1 02 8d cb  cf
8b90 : 02 20 0c c5 ad c2 02 8d  72
8b98 : cb 02 20 0c c5 ad c3 02  cb
8ba0 : 8d cb 02 20 0c c5 ad c0  bf
8ba8 : 02 ac c1 02 20 f7 c5 ad  e5
8bb0 : c2 02 ac c3 02 4c f7 c5  05
8bb8 : ad c0 02 8d cb 02 20 54  ee
8bc0 : c5 ad c1 02 8d cb 02 20  8c
8bc8 : 54 c5 ad c2 02 8d cb 02  82
8bd0 : 20 54 c5 ad c3 02 8d cb  5b
8bd8 : 02 20 54 c5 ad c0 02 ac  fa
8be0 : c2 02 20 f7 c5 ad c1 02  7f
8be8 : ac c3 02 4c f7 c5 ad c0  66
8bf0 : 02 8d cb 02 20 7f c5 ad  5c
8bf8 : c1 02 8d cb 02 20 7f c5  42
8c00 : ad c2 02 8d cb 02 20 7f  8d
8c08 : c5 ad c3 02 8d cb 02 20  54
8c10 : 7f c5 ad c0 02 ac c1 02  86
8c18 : 20 f7 c5 ad c1 02 ac c2  bf
8c20 : 02 20 f7 c5 ad c1 02 ac  33
8c28 : c3 02 4c f7 c5 ad 00 d0  6a
8c30 : c9 68 b0 15 ad 01 d0 c9  b6
8c38 : 7a b0 07 ad c0 02 8d cb  6c
8c40 : 02 60 ad c2 02 8d cb 02  f6
8c48 : 60 ad 01 d0 c9 7a b0 07  1a
8c50 : ad c1 02 8d cb 02 60 ad  ba
8c58 : c3 02 8d cb 02 60 ea a9  1b
8c60 : 37 85 01 a9 00 a2 04 85  00
8c68 : f9 86 fa a9 8a a2 93 85  af
8c70 : f7 86 f8 20 65 c7 a9 00  28
8c78 : a2 d8 85 f9 86 fa a9 8a  23
8c80 : a2 97 85 f7 86 f8 20 65  ca
8c88 : c7 a9 37 85 01 58 60 a2  3c
8c90 : 04 a0 00 b1 f7 91 f9 c8  a0
8c98 : d0 f9 e6 f8 e6 fa ca [illegible]  50
8ca0 : f2 60 ac cb 02 20 51 c0  4f
8ca8 : a0 00 a9 00 91 f7 c8 c0  30
8cb0 : 08 d0 f9 60 48 ad 86 02  bb
8cb8 : 29 07 85 f7 ad 21 d0 29  3f
8cc0 : 0f 85 f8 ad 22 d0 29 0f  f1
8cc8 : 85 f9 ad 23 d0 29 0f 85  b7
8cd0 : fa 68 c9 01 f0 42 c9 00  d9
8cd8 : f0 1d a5 fa aa e8 8a c9  cf
8ce0 : 10 d0 02 a9 00 c5 f7 f0  fe
8ce8 : f3 c5 f8 f0 ef c5 f9 f0  11
8cf0 : eb 8d 23 d0 4c 08 c8 a5  f8
8cf8 : f7 aa e8 8a c9 08 d0 02  f4
8d00 : a9 00 c5 f9 f0 f3 c5 fa  16
8d08 : f0 ef c5 f8 f0 eb 09 08  23
8d10 : 8d 86 02 29 07 4c 08 c8  0b
8d18 : a5 f9 aa e8 8a c9 10 d0  5a
8d20 : 02 a9 00 c5 f7 f0 f3 c5  12
8d28 : fa f0 ef c5 f8 f0 eb 8d  31
8d30 : 22 d0 8d 0c d9 ad 86 02  c8
8d38 : 29 07 ae 22 d0 ac 23 d0  75
8d40 : 8d 6c d8 8e 94 d8 8c bc  c7
8d48 : d8 20 7b c3 4c ef c2 ad  32
8d50 : c4 02 8d 16 d0 4c bc fe  9c
8d58 : a9 c8 8d 16 d0 4c bc fe  ec
8d60 : a9 06 8d 15 d0 ad 86 02  ab
8d68 : 85 ff a9 07 8d 86 02 a9  a0
8d70 : 20 8d a7 06 a2 10 a0 14  57
8d78 : 18 20 0a e5 a2 00 bd f8  f3
8d80 : c8 20 d2 ff e8 e0 05 d0  58
8d88 : f5 a9 00 85 cc a2 00 a9  38
8d90 : 00 85 c6 a5 c6 f0 fc ad  fc
8d98 : 77 02 c9 14 f0 1e c9 20  6d
8da0 : f0 0c c9 0d f0 21 c9 23  30
8da8 : 30 e5 c9 5f 10 e1 e0 0d  d7
8db0 : b0 dd 20 d2 ff 9d 0a c9  5a
8db8 : e8 4c 65 c8 e0 01 30 cf  af
8dc0 : ca 20 d2 ff 4c 65 c8 20  a2
8dc8 : ac c9 a9 00 85 c6 e8 e8  c7
8dd0 : e8 8a 48 a9 01 85 cc a9  07
8dd8 : 42 8d a7 06 a9 0f 8d a7  24
8de0 : da a9 00 85 9d a9 01 a0  ac
8de8 : c9 a2 1a 20 bd ff a9 01  11
8df0 : a2 08 a0 0f 20 ba ff 20  b8
8df8 : c0 ff 90 0a a9 01 20 c3  c8
8e00 : ff 20 c2 c9 68 60 a9 01  2b
8e08 : 20 c3 ff ad 11 d0 29 ef  db
```

```
8e10 : 8d 11 d0 68 a2 07 a0 c9  df
8e18 : 20 bd ff a5 fe f0 3f 4c  d8
8e20 : 1b c9 4e 41 4d 45 3a 44  4c
8e28 : 49 53 4b 20 45 52 52 4f  c1
8e30 : 52 40 3a 13 20 20 20 20  57
8e38 : 20 20 20 20 20 20 20 20  38
8e40 : 20 20 20 20 20 a2 08 20  f4
8e48 : ba ff a2 00 a9 30 86 f7  d1
8e50 : 85 f8 a9 f7 a2 00 a0 40  e8
8e58 : 20 d8 ff 4c 47 c9 a2 08  cb
8e60 : a0 00 20 ba ff a9 00 a2  f2
8e68 : 00 a0 30 20 d5 ff 20 cd  42
8e70 : c9 20 c2 c9 a9 07 8d 86  49
8e78 : 02 a9 08 85 fa 20 b4 ff  85
8e80 : a9 6f 85 b9 20 96 ff 20  70
8e88 : a5 ff c9 30 d0 0a 20 a5  ce
8e90 : ff c9 30 d0 03 4c 8f c9  fe
8e98 : a2 10 a0 14 18 20 0a e5  63
8ea0 : a2 00 bd fd c8 20 d2 ff  4a
8ea8 : e8 e0 0a d0 f5 20 8f c9  cf
8eb0 : a5 cb c9 3c d0 fa 4c c2  d0
8eb8 : c9 20 a5 ff c9 0d d0 f9  37
8ec0 : 20 ab ff a5 ff 8d 86 02  f5
8ec8 : ad 11 d0 09 10 8d 11 d0  a6
8ed0 : a9 07 8d 15 d0 60 a9 20  fa
8ed8 : 9d 9a 06 bd 0a c9 c9 a0  53
8ee0 : f0 04 c9 20 d0 04 ca 4c  3a
8ee8 : ac c9 e8 60 a9 20 a2 13  0b
8ef0 : 9d 93 06 ca d0 fa 60 a9  eb
8ef8 : 00 a2 30 a0 40 85 f7 85  85
8f00 : f9 86 f8 84 fa a8 b1 f7  b7
8f08 : 91 f9 c8 d0 f9 e6 f8 e6  6a
8f10 : fa a5 f8 c9 40 d0 ef 60  5f
8f18 : 20 03 c7 ac cb 02 20 51  31
8f20 : c0 ad 01 d0 38 e9 3d 4a  6e
8f28 : 4a 4a 29 07 8d cb 02 60  c3
8f30 : 20 ee c9 4c 12 ca 20 ee  99
8f38 : c9 4c 4c ca a0 07 b1 f7  8d
8f40 : ee 00 7e d0 05 a2 01 8e  6f
8f48 : 00 7e ae 00 7e 9d 00 7e  05
8f50 : ad cb 02 c9 07 f0 0c 88  d6
8f58 : b1 f7 c8 91 f7 88 cc cb  f8
8f60 : 02 d0 f4 ae 00 7f bd 00  d0
8f68 : 7f e0 00 f0 03 ce 00 7f  1b
8f70 : ac cb 02 91 f7 60 ac cb  81
8f78 : 02 b1 f7 ee 00 7f d0 05  78
8f80 : a2 01 8e 00 7f ae 00 7f  b3
8f88 : 9d 00 7f c0 07 f0 0b c8  d3
8f90 : b1 f7 88 91 f7 c8 c0 07  68
8f98 : d0 f5 ae 00 7e bd 00 7e  e1
8fa0 : e0 00 f0 03 ce 00 7e a0  45
8fa8 : 07 91 f7 60 ad 01 d0 38  18
8fb0 : e9 3d 4a 4a 4a 8d cc 02  5c
8fb8 : 29 07 8d cb 02 ad 00 d0  71
8fc0 : c9 65 b0 17 ac c0 02 20  64
8fc8 : 51 c0 85 f9 86 fa ac c2  92
8fd0 : 02 20 51 c0 85 fb 86 fc  9b
8fd8 : 4c c5 ca ac c1 02 20 51  9e
8fe0 : c0 85 f9 86 fa ac c3 02  da
8fe8 : 20 51 c0 85 fb 86 fc ad  d5
8ff0 : cc 02 29 08 60 20 82 ca  af
8ff8 : d0 22 a5 f9 a6 fa 85 f7  ca
9000 : 86 f8 20 12 ca ae 00 7e  6c
9008 : bd 00 7e ce 00 7e ee 00  ee
9010 : 7f ae 00 7f 9d 00 7f a9  01
9018 : 00 8d cb 02 a5 fb a6 fc  e1
9020 : 85 f7 86 f8 20 12 ca 60  e0
9028 : 20 82 ca d0 05 a9 00 8d  0f
9030 : cb 02 a5 fb a6 fc 85 f7  3d
9038 : 86 f8 20 4c ca ad cc 02  1d
9040 : 29 08 d0 25 ae 00 7f bd  aa
9048 : 00 7f ce 00 7f ee 00 7e  28
9050 : ae 00 7e 9d 00 7e ad cc  96
9058 : 02 29 07 8d cb 02 a5 f9  ba
9060 : a6 fa 85 f7 86 f8 20 4c  2d
9068 : ca 60 a9 06 8d 15 d0 ad  ae
9070 : 86 02 85 ff a9 07 8d 86  6f
9078 : 02 78 a9 00 8d 19 d0 8d  21
9080 : 1a d0 a9 31 a2 ea 8d 14  73
9088 : 03 8e 15 03 ad 16 d0 85  52
9090 : fe a9 15 8d 18 d0 a9 c8  9a
9098 : 8d 16 d0 58 a2 10 a0 14  c5
90a0 : 18 20 0a e5 a9 13 a2 cc  5f
90a8 : 85 bb 86 bc a9 04 85 b7  84
90b0 : a9 08 85 ba a9 60 85 b9  3d
90b8 : 20 d5 f3 a5 ba 20 b4 ff  f4
90c0 : a5 b9 20 96 ff a9 00 85  75
90c8 : 90 a0 03 84 fb 20 a5 ff  51
90d0 : a4 90 d0 54 20 a5 ff a4  f3
90d8 : 90 d0 4d a4 fb 88 d0 eb  d7
90e0 : 20 a5 ff a6 90 d0 41 aa  91
90e8 : c9 22 f0 2b 4c b6 cb a2  53
90f0 : 10 a0 14 18 20 0a e5 a5  8e
90f8 : c5 ae 8d 02 e0 01 d0 06  1d
9100 : 20 17 cc 4c fe cb c9 40  5e
9108 : f0 ed a5 c5 c9 40 d0 fa  e9
9110 : 20 17 cc a0 02 d0 b4 20  bd
9118 : a5 ff c9 22 f0 d1 c9 13  5e
```

Listing 3. Charakter-Designer Hauptprogramm (Fortsetzung). Bei gleicher Eingabe bitte den MSE verwenden.

```
9120 : f0 f5 20 d2 ff 4c ed cb  1f
9128 : 20 42 f6 a5 ff 8d 86 02  66
9130 : a9 00 85 c6 20 00 c0 a5  64
9138 : fe 8d 16 d0 60 24 3a 13  d3
9140 : 2a a9 20 a2 12 9d 93 06  04
9148 : ca d0 fa 60 ad 86 02 85  67
9150 : ff a9 06 8d 15 d0 4c a6  ad
9158 : cc a2 13 a0 01 18 20 0a  b4
9160 : e5 a2 13 a0 01 a9 00 85  d8
9168 : c6 a5 c6 f0 fc ad 77 02  f0
9170 : c9 14 f0 2f c9 20 f0 18  f7
9178 : c9 0d f0 61 c9 91 f0 40  9d
9180 : c9 11 f0 44 c9 1d f0 08  f0
9188 : c9 9d f0 17 c9 93 f0 40  bc
9190 : c0 27 b0 04 c8 4c 8c cc  4b
9198 : a0 01 18 20 0a e5 a9 11  5c
91a0 : 4c 9e cc c0 02 30 04 88  49
91a8 : 4c 8c cc a0 26 18 20 0a  39
91b0 : e5 a9 91 4c 96 cc 20 d2  4e
91b8 : ff 18 20 0a e5 4c 3b cc  54
91c0 : e0 14 30 a1 ca 4c 8c cc  c5
91c8 : e0 17 10 99 e8 4c 8c cc  28
91d0 : a2 00 a9 20 9d d0 06 e8  2b
91d8 : d0 fa 4c 2f cc a2 00 86  0d
91e0 : c6 a5 ff 8d 86 02 a9 07  58
91e8 : 8d 15 d0 4c 7b c3 a9 00  3a
91f0 : a2 38 a0 30 85 f7 85 f9  ff
91f8 : 86 f8 84 fa a8 b1 f7 aa  c8
9200 : b1 f9 91 f7 8a 91 f9 c8  c0
9208 : d0 f3 e6 f8 e6 fa a5 fa  7d
9210 : c9 38 d0 e9 60 ad 01 d0  80
9218 : c9 7d b0 13 ac c1 02 20  4f
9220 : 51 c0 85 f9 86 fa ac c0  e6
9228 : 02 20 51 c0 4c 15 cd ac  a5
9230 : c3 02 20 51 c0 85 f9 86  54
9238 : fa ac c2 02 20 51 c0 ad  64
9240 : 01 d0 38 e9 3d 4a 4a 4a  d8
9248 : 29 07 8d cb 02 a5 f7 18  2f
9250 : 6d cb 02 85 f7 a5 f9 18  99
9258 : 6d cb 02 85 f9 ad 00 d0  8b
9260 : c9 6c 08 38 e9 2c 4a 4a  26
9268 : 4a 29 07 a8 a9 00 8d cb  86
9270 : 02 c0 00 f0 08 38 6a 88  ed
9278 : d0 fb 8d cb 02 28 60 20  46
9280 : eb cc 90 08 a5 f9 a6 fa  b1
9288 : 85 f7 86 f8 a0 00 b1 f7  8a
9290 : 2d cb 02 85 fb ad cb 02  34
9298 : 49 ff 8d cb 02 b1 f7 2d  a5
92a0 : cb 02 4a 05 fb 91 f7 60  8c
92a8 : 20 eb cc 90 08 a5 f9 a6  e6
92b0 : fa 85 f7 86 f8 a0 00 b1  33
92b8 : f7 2d cb 02 85 fb ad cb  ff
92c0 : 02 49 ff 8d cb 02 b1 f7  9c
92c8 : 0a 2d cb 02 05 fb 91 f7  02
92d0 : 60 20 eb cc 90 03 4c 55  d2
92d8 : cd a0 00 b1 f7 29 01 85  03
92e0 : 02 20 55 cd a0 00 b1 f9  c6
92e8 : a6 02 e0 01 6a 91 f9 60  c3
92f0 : 20 eb cc 90 03 4c 7e cd  73
92f8 : a0 00 b1 f9 18 0a 90 07  66
9300 : a2 01 86 02 4c e1 cd a2  55
9308 : 00 86 02 91 f9 20 7e cd  34
9310 : a5 02 f0 08 a0 00 b1 f7  b4
9318 : 09 01 91 f7 60 a9 06 8d  8c
9320 : 15 d0 ad 86 02 85 ff a9  79
9328 : 07 8d 86 02 a2 10 a0 14  2d
9330 : 18 20 0a e5 a0 0d b9 45  7b
9338 : ce 20 d2 ff 88 d0 f7 a5  05
9340 : cb c9 40 f0 fa c9 3e d0  b7
9348 : 34 a5 cb c9 40 d0 fa c8  83
9350 : c0 03 d0 eb 78 a9 f3 a2  2d
9358 : 70 85 f7 86 f8 a0 00 98  1f
9360 : 91 f7 c8 d0 fb e6 f8 a5  5f
9368 : f8 c9 cd d0 f3 4c e2 fc  f9
9370 : 54 45 53 45 52 20 47 4e  c4
9378 : 55 54 48 43 41 a5 ff 8d  ce
9380 : 86 02 a9 07 8d 15 d0 4c  b0
9388 : c2 c9 20 55 40 40 40 40  69
9390 : 40 40 40 7d 6d 40 40 40  0a
9398 : 40 40 40 40 49 60 20 20  69
93a0 : 20 20 20 20 20 20 20 20  a0
93a8 : 55 40 40 40 40 40 40 40  bd
93b0 : 40 49 31 42 20 20 20 20  ed
93b8 : 20 20 20 20 20 20 20 20  b8
93c0 : 20 20 20 20 42 55 40 40  4d
93c8 : 40 40 49 55 40 40 40 49  bf
93d0 : 42 1e 20 2d 20 05 0e 04  3a
93d8 : 05 42 32 42 20 20 20 20  97
93e0 : 60 20 20 20 20 20 20 20  20
93e8 : 20 20 20 20 42 42 20 20  1b
93f0 : 20 20 42 42 b1 20 51 42  df
93f8 : 42 00 20 2d 20 09 0e 09  7d
9400 : 14 42 33 42 20 20 20 20  0e
9408 : 20 20 20 20 20 20 20 20  08
9410 : 20 20 20 20 42 42 20 20  43
9418 : 20 20 42 42 b2 20 51 42  17
9420 : 42 0c 20 2d 20 0c 0f 01  b7
9428 : 04 42 34 42 20 20 20 20  67
9430 : 20 20 20 20 20 20 20 20  30
9438 : 20 20 20 20 42 42 20 20  6b
9440 : 20 60 42 42 b3 20 51 42  6f
9448 : 42 13 20 2d 20 13 01 16  8c
9450 : 05 42 35 42 20 20 20 20  d0
9458 : 20 20 20 20 20 20 20 20  58
9460 : 20 20 20 20 42 42 20 20  93
9468 : 20 20 42 42 b4 20 d1 42  89
9470 : 42 14 20 2d 20 14 05 13  47
9478 : 14 42 36 42 20 20 20 20  47
9480 : 20 20 20 20 20 20 20 20  80
9488 : 20 20 20 20 42 4a 40 40  bc
9490 : 40 40 4b 42 01 20 51 42  e6
9498 : 42 09 20 2d 20 09 0e 16  bb
94a0 : 05 42 37 42 20 20 20 20  a0
94a8 : 20 20 20 20 20 20 20 20  a8
94b0 : 20 20 20 20 42 20 55 40  e7
94b8 : 40 49 20 4a 40 40 40 4b  8c
94c0 : 42 0b 20 2d 20 0b 0f 10  ec
94c8 : 06 42 38 7d 20 20 20 20  71
94d0 : 20 20 20 20 20 20 20 20  d0
94d8 : 20 20 20 20 6d 20 42 20  36
94e0 : 20 42 20 55 40 40 40 40  5c
94e8 : 4b 03 20 2d 20 03 0f 10  d9
94f0 : 19 42 31 6e 20 20 20 20  08
94f8 : 20 20 20 20 20 20 20 20  f8
9500 : 20 20 20 20 70 20 42 20  8e
9508 : 20 42 20 42 04 20 2d 20  d0
9510 : 04 12 05 08 05 0e 20 20  e1
9518 : 20 42 32 42 20 20 20 20  f2
9520 : 20 20 20 20 20 20 20 20  20
9528 : 20 20 20 20 42 20 4a 40  33
9530 : 40 4b 20 42 06 20 2d 20  bd
9538 : 13 10 09 05 07 05 0c 0e  1b
9540 : 20 42 33 42 20 20 20 20  5a
9548 : 20 20 20 20 20 20 20 20  48
9550 : 20 20 20 20 42 20 20 20  72
9558 : 20 20 20 42 07 20 2d 20  3f
9560 : 07 12 0f 13 13 2f 0b 0c  86
9568 : 2e 42 34 42 20 20 20 20  d1
9570 : 20 20 20 20 20 20 20 20  70
9578 : 20 20 20 20 42 60 55 40  b1
9580 : 40 40 40 4b 1a 20 2d 20  f1
9588 : 1a 15 12 15 05 03 0b 20  29
9590 : 20 42 35 42 20 20 20 20  2b
9598 : 20 20 20 20 20 20 20 20  98
95a0 : 20 20 20 20 42 60 42 20  4d
95a8 : 09 0e 13 2f 1c 20 2d 20  1b
95b0 : 05 09 0e 06 15 05 07 05  1e
95b8 : 0e 42 36 42 20 20 20 20  81
95c0 : 20 20 20 20 20 20 20 20  c0
95c8 : 20 20 20 20 42 20 42 03  39
95d0 : 0f 0d 0d 0f 04 2e 2d 20  32
95d8 : 17 01 05 08 0c 05 0e 55  7e
95e0 : 40 4b 37 42 20 20 20 20  a0
95e8 : 20 20 20 20 20 20 20 20  e8
95f0 : 20 20 20 20 4[illegible] [illegible] 71 40  42
95f8 : 40 40 40 40 40 40 40 40  f8
9600 : 40 40 40 40 40 40 40 71  62
9608 : 40 49 38 42 20 20 20 20  07
9610 : 20 20 20 20 20 20 20 20  10
9618 : 20 20 20 20 42 42 20 20  4b
9620 : 20 20 20 20 20 20 20 20  20
9628 : 20 20 20 20 20 20 20 60  a9
9630 : 20 42 20 4a 40 40 40 40  4a
9638 : 40 40 40 6e 70 40 40 40  01
9640 : 40 40 40 40 4b 4a 40 40  41
9648 : 40 40 40 40 40 40 40 40  48
9650 : 40 40 40 40 40 40 40 40  50
9658 : 40 4b 20 20 20 20 20 20  0e
9660 : 20 20 20 20 20 20 20 20  60
9668 : 20 20 20 20 20 20 20 20  68
9670 : 20 20 20 20 20 20 20 20  70
9678 : 20 20 20 20 20 20 20 20  78
9680 : 20 20 20 20 20 20 20 20  80
9688 : 20 20 20 20 20 20 20 20  88
9690 : 20 20 20 20 20 20 20 20  90
9698 : 20 20 20 20 20 20 20 20  98
96a0 : 20 20 20 20 20 20 20 20  a0
96a8 : 20 20 60 20 20 20 20 20  b8
96b0 : 20 20 20 20 20 20 20 20  b0
96b8 : 20 20 20 20 20 20 20 20  b8
96c0 : 20 20 20 20 20 20 20 20  c0
96c8 : 20 20 20 20 20 20 20 20  c8
96d0 : 20 20 60 20 20 20 20 20  e0
96d8 : 20 20 20 20 20 20 20 20  d8
96e0 : 20 20 20 20 20 20 20 20  e0
96e8 : 20 20 20 20 20 20 20 20  e8
96f0 : 20 20 20 20 20 20 20 20  f0
96f8 : 20 20 60 20 20 20 20 20  08
9700 : 20 20 20 20 20 20 20 20  00
9708 : 20 20 20 20 20 20 20 20  08
9710 : 20 20 20 20 20 20 20 20  10
9718 : 20 20 20 20 20 20 20 20  18
9720 : 20 20 60 20 20 20 20 20  30
9728 : 20 20 20 20 20 20 20 20  28
9730 : 20 20 20 20 20 20 20 20  30
9738 : 20 20 20 20 20 20 20 20  38
9740 : 20 20 20 20 20 20 20 20  40
9748 : 20 20 60 20 20 20 20 20  58
9750 : 20 20 20 20 20 20 20 20  50
9758 : 20 20 20 20 20 20 20 20  58
9760 : 20 20 20 20 20 20 20 20  60
9768 : 20 20 20 20 20 20 20 20  68
9770 : 20 20 00 00 00 00 00 00  a1
9778 : 00 00 00 00 00 00 00 00  79
9780 : 00 00 00 00 00 00 00 00  81
9788 : 00 00 f0 0f ff 5f ff ff  a1
9790 : 2f 4f 2f ff ff 5f ff 9f  6d
9798 : 0f 8f ff cf ff 41 f7 f7  42
97a0 : f7 07 f7 17 f7 f7 d7 57  49
97a8 : ff cf ff ff 5f 1f ff ff  7e
97b0 : ff 9f f0 2f 2f ff ff ff  94
97b8 : 0f bf bf 0f 4f af 9f af  c9
97c0 : ff ff ff 5f ff ff ff ff  ab
97c8 : 2f ff ff f7 ff 9f ff 9f  32
97d0 : ff f1 5f 20 2f f7 f7 07  44
97d8 : f7 9f 60 ff b0 c0 20 f0  2a
97e0 : f0 ff 9f 4f 5f ff ff 5f  56
97e8 : ff 5f af ff ff 1f 4f 5f  78
97f0 : ff ff 5f bf f1 9f f3 4f  49
97f8 : 0f b1 f1 b0 f0 c7 27 f7  4c
9800 : f7 df 90 4f 5f 2f ff ff  64
9808 : 9f ff cf ff ff cf 0f 5f  14
9810 : 4f ff af 0f 4f 5f ff 9f  5c
9818 : ff bf af bf f1 ff 67 6f  76
9820 : 4f 41 07 f0 f7 c7 c7 f7  bc
9828 : b7 ff 40 6f 8f 9f ff ff  d3
9830 : 0f 9f ff 4f ff ff 0f ff  35
9838 : 3f ff ff ff 4f 2f 4f 0f  40
9840 : 9f ff 4f 9f 51 ff f0 ff  7f
9848 : ff 81 27 f0 f7 47 07 e7  95
9850 : 07 ff f0 bf 9f 0f 4f ff  3b
9858 : ff ff 5f 4f 0f 1f ff 4f  a1
9860 : ff ff 1f ff ff ff ff af  86
9868 : 8f ff ff ff f1 af f1 ff  5b
9870 : ff f1 f0 f0 f7 f7 57 47  ed
9878 : 57 cf f0 ff ff ff 1f ff  6f
9880 : 6f ff cf ff cf af ff 5f  1c
9888 : ff 9f 1f ff df ff ff ff  1d
9890 : 5f ff ff ff b1 f8 93 ff  20
9898 : 9f c1 b1 40 f1 f7 57 07  d6
98a0 : 57 5f f0 ff 4f 4f 4f af  ef
98a8 : ff ff 4f ef df ff df ff  f6
98b0 : 2f 5f ff ff ff 5f 9f 0f  26
98b8 : ff ff ff cf 5f 5f ff ff  a2
98c0 : ff 51 b7 10 f7 f7 f7 87  86
98c8 : c7 5f b0 ff ff 4f 5f 5f  22
98d0 : ff ff cf 5f bf 9f ff ff  a8
98d8 : 4f 6f 4f ff ff 5f 0f 2f  48
98e0 : 0f ff ff ff ff ef ef ff  2e
98e8 : ff 41 a7 f0 97 17 f7 b7  11
98f0 : f7 4f f0 ff 11 91 51 01  b0
98f8 : d1 f1 41 41 41 f1 f1 f1  8a
9900 : 41 b1 41 af ff af 5f ff  5b
9908 : ff ff ff 2f 91 07 f0 97  31
9910 : b7 07 57 57 07 f7 ff ff  3c
9918 : bf ff 60 ff ff ff 4f 4f  cb
9920 : ff ff 01 ff 4f 0f ff ff  cd
9928 : 5f ff 9f 0f 9f ff 4f 6f  67
9930 : 1f 4f ff ff f1 f7 f0 27  e7
9938 : f7 f7 d7 57 57 f7 f7 f7  11
9940 : ff 2f 20 ff ff 2f 9f 1f  15
9948 : 5f 0f f1 5f 0f 0f 0f ff  3d
9950 : ff 4f df af ff ff ff cf  84
9958 : 5f ff ff ff 51 97 10 27  17
9960 : f7 f7 47 27 57 a7 f7 f7  8c
9968 : 5f df f0 ff ff ff 6f 1f  ef
9970 : 4f ff f1 ff ff bf ff bf  b9
9978 : ff 4f af ff ff ef bf cf  29
9980 : 0f 2f ff ff 41 f1 d0 20  4e
9988 : f7 f7 57 07 b7 c7 f7 f7  bb
9990 : 07 cf 00 ff ff 9f 21 0f  1e
9998 : cf f1 ff bf cf 1f 1f ff  ca
99a0 : ff bf 2f 0f 5f ff ff 27  71
99a8 : a1 a1 f1 91 c1 51 10 a0  f0
99b0 : f7 f7 47 47 f7 b7 f7 f7  6b
99b8 : 07 9f f0 ff ff ff cf 9f  49
99c0 : 0f ff ff ff 8f ef ff ff  47
99c8 : ff cf 9f 5f ff 1f 9f 41  7d
99d0 : 41 f1 41 f1 f1 c1 00 f7  b5
99d8 : f7 97 27 67 f7 f7 b7 ff  70
99e0 : ff cf d0 ff ff 4f 4f ff  b3
99e8 : ff ff 1f 9f 9f ff ff ff  9d
99f0 : 5f 5f 1f 1f df ff 4f ff  e6
99f8 : cf ff ff ff af 4f 9f ff  bb
9a00 : ff bf 4f ff 2f ff ff 2f  04
9a08 : 5f ff f0 ff ff cf 9f 0f  be
9a10 : ff ff ff 4f ff 2f ff ff  72
9a18 : 2f 5f bf 4f ff ff 0f 0f  2b
9a20 : bf ff ff 4f ef df 2f 9f  c3
9a28 : ff 5f ff 5f ff 0f 0f df  37
9a30 : 1f 5f ff 1f ff 4f 4f ff  9a
9a38 : ff ff 1f 5f 5f ff ff ff  e1
9a40 : ff cf bf ff 9f 2f 0f cf  66
9a48 : 5f ff af 4f ff 2f 5f 1f  b2
9a50 : ff 5f 5f bf ff ff 2f 9f  cb
9a58 : ff 0f ff bf ff af 0f 0f  ae
9a60 : ff df 5f 5f ff ff ff ff  13
9a68 : 6f ff cf ff ff 4f 9f 6f  a2
9a70 : 0f df ff 4f 0f df ff ff  49
9a78 : 4f 6f ff cf ff ff 4f ff  b6
9a80 : 2f 0f 1f ff ff ff ff ff  ff
```

Listing 3. Charakter-Designer Hauptprogramm (Schluß). Bei der Eingabe bitte den MSE verwenden.

3D-Supergrafik

Ein dreidimensionales Grafikprogramm, das sich auch vor professionellen 3D-Programmen nicht zu verstecken braucht. Bewiesen wurde das, indem dieses Programm im Rahmen des Wettbewerbs »Jugend forscht '85« mit dem Sonderpreis Elektronik ausgezeichnet wurde.

Mit diesem Programm ist es möglich, beliebige dreidimensionale Körper zu drehen und in verschiedenen Perspektiven darzustellen. Die Besonderheit dieses Programms ist jedoch, daß es eine Routine zur Lösung des Hinterschneidungsproblems besitzt. Diese Routine ermöglicht es, Körper naturgetreu darzustellen, indem sie Kanten, die von dem Körper verdeckt werden, löscht. Außer der Darstellung von Körpern besitzt das Programm natürlich auch Routinen zum Eingeben, Laden, Speichern und Ausdrucken von Körpern sowie eine eigene Grafikerweiterung in Maschinensprache.

Insgesamt ist das Programm sehr übersichtlich und leicht zu bedienen. Nur die Routine zur Eingabe von Körpern erfordert eine kurze Einarbeitungszeit. Denn zuerst müssen die Eckpunkt-Koordinaten eingegeben werden, dann die Nummern der Punkte, aus denen die Fläche besteht, und zwar in der Reihenfolge, in der sie verbunden werden sollen.

Die Routine zur Berechnung des Bildes erlaubt keine Fehleingaben, so daß man durch Probieren die Auswirkungen der einzelnen Faktoren auf das Bild bestimmen kann. Außerdem braucht sich der Benutzer nicht um die Größe der Darstellung zu kümmern, da dies vom Computer erledigt wird.

Die Speicherbelegung

Bei 3D-Supergrafik handelt es sich eigentlich um drei Programme: Einem Lader (Listing 1), eine Grafikerweiterung in Maschinensprache (Listing 3) und das Hauptprogramm in Basic (Listing 2). Diese Grafikerweiterung belegt den Speicherplatz von $C400 bis $CAFF. Außerdem benutzt sie zwei Grafikbildschirme unter den beiden ROM-Bereichen ($A000-$BFFF/$E000-$FFFF). Die dazugehörigen Farbspeicher liegen von $8C00 bis $8FFF und von $C000 bis $C3FF. Daraus folgt, daß das Hauptprogramm mit den Variablen im Bereich von $0800 bis $8BFF liegt. Soviel zur Speicherbelegung.

Das Programm

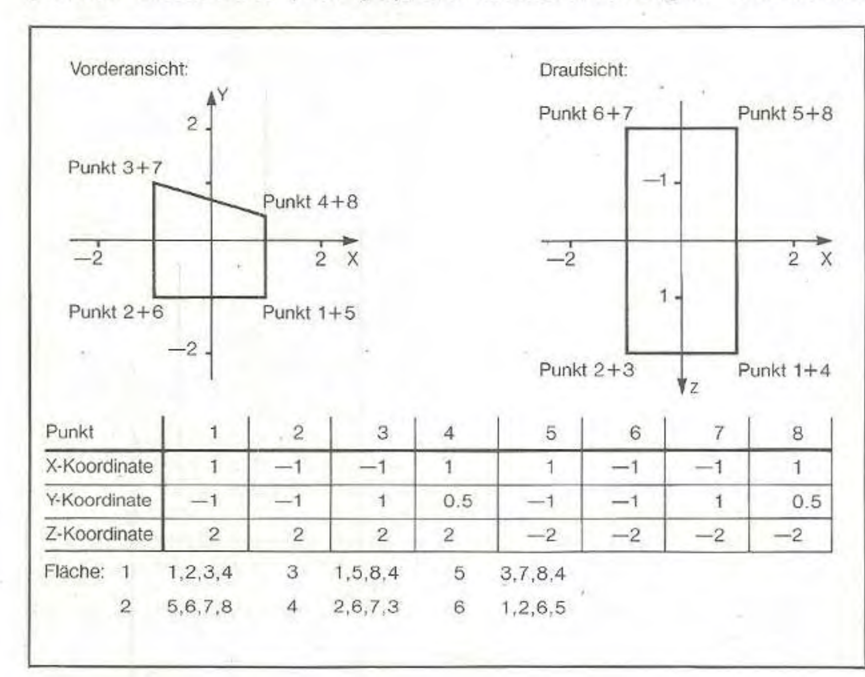
Nach Laden und Starten des Laders gelangt man nach dreißig Sekunden ins Hauptmenü, das fünf Menüpunkte zur Verfügung stellt. Auf die Bedienung der einzelnen Punkte werde ich im folgenden eingehen.

1. Daten eingeben

Bevor man einen dreidimensionalen Körper in den Computer eingeben kann, muß man ihn erst in mathematische Daten zerlegen. Das hört sich schwerer an als es eigentlich ist. Man zeichnet sich einfach eine Vorderansicht und eine Draufsicht des Körpers in Koordinatensysteme. Daraus kann man dann die Eckpunktkoordinaten und die Verbindungsvorschrift ablesen (siehe Bild 1). Nachdem Sie diese Arbeit erledigt haben, können Sie mit der Eingabe der Daten in den Computer beginnen.

Zuerst muß man die Anzahl der Eckpunkte (im Bild 1: 8 Punkte) und die Anzahl der Flächen (im Bild 1: 6 Flächen) des gewünschten Körpers eingeben. Danach sind die räumlichen Koordinaten der Eckpunkte an der Reihe. Nachdem man die Koordinaten des Punktes eingegeben hat, wird man vom Computer gefragt, ob die Eingabe richtig ist. Falls dies so ist, tippt man einfach weiter, falls nicht, drückt man die Taste »N«. Dann kann man die Werte noch einmal eingeben.

Nachdem man alle Eckpunkte definiert hat, muß man eingeben, in welcher Reihenfolge die Punkte zu jeder Fläche verbunden werden sollen. Ist man mit einer Fläche fertig, gibt man nur RETURN ein. Jetzt folgt wieder die Richtigkeitsabfrage des Computers, die analog zur vorherigen Frage beantwortet werden muß. Leider gibt es bei der Flächeneingabe noch eine Einschränkung. Eine Fläche darf nämlich nur aus vier Eckpunkten bestehen. Daraus folgt, daß große Flächen zerlegt, und auf mehrere kleine Flächen verteilt werden müssen (siehe Bild 2). Durch diese Maßnahme entstehen Linien, die die große Fläche teilen. Um so etwas zu vermeiden, kann der Anwender Linien eingeben, die später im Bild nicht gezeichnet werden. Das geht folgendermaßen:



Punkt	1	2	3	4	5	6	7	8
X-Koordinate	1	−1	−1	1	1	−1	−1	1
Y-Koordinate	−1	−1	1	0.5	−1	−1	1	0.5
Z-Koordinate	2	2	2	2	−2	−2	−2	−2

Fläche:					
1	1,2,3,4	3	1,5,8,4	5	3,7,8,4
2	5,6,7,8	4	2,6,7,3	6	1,2,6,5

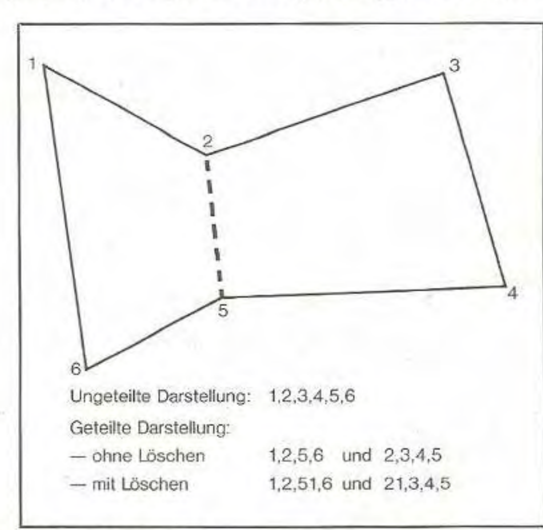


Bild 2. Beispiel für die Teilung einer »großen« Fläche

◄ Bild 1. Beispiel für die Erstellung eines Körpers

Man schreibt hinter den Punkt, **zu** dem die nachher nicht sichtbare Linie führen soll, ein »l«.

So, das war das Schlimmste. Alles weitere ist einfach.

2. Daten laden

Dieser Menüpunkt dient dazu, die Daten eines Körpers von Diskette zu laden. Er bietet noch zwei Sonderfunktionen:
»$« druckt das Directory der Datenfiles auf der Disk
»@« Rückkehr zum Hauptmenü

Bei der Angabe, welches Bild zu laden ist, darf das »3D-« nicht mit eingegeben werden.

3. Daten speichern

Diesen Menüpunkt sollte man sofort nach der Eingabe eines Körpers aufrufen, um die Daten zu sichern. Er speichert die Daten in einem platzsparendem Format auf Disk. Die Sonderfunktionen sind analog zu Punkt 2. Dem Dateinamen wird automatisch ein »3D-« vorangestellt.

4. Bild berechnen

Es handelt sich hierbei um den umfangreichsten Punkt des Programmes, der aber nicht schwer zu verstehen ist. Die Auswirkung der einzelnen Parameter auf das Bild läßt sich am besten durch folgende Methode herausfinden: Probieren, probieren, probieren ...

Für Interessierte möchte ich hier noch erwähnen, wie ich das Hinterschneidungsproblem in meinem Programm gelöst habe:

Der Computer zeichnet die hinterste Fläche zuerst und füllt sie im unsichtbaren Grafikschirm aus (zu sehen, wenn Sie »F2« drücken). Jeden Punkt, den er dort setzt, löscht er in der sichtbaren Grafik. Auf diese Weise verfährt er auch mit den anderen Flächen, die vorderste Fläche zeichnet er zuletzt. Diese Verfahrensweise läßt sich sehr gut beim Bildaufbau beobachten. Es sieht aus, als ob der Computer den Körper aus einzelnen Teilen zusammensetzt. Es lohnt sich auch, mit den Funktionstasten F1 und F2 beim Bildaufbau zwischen den beiden Grafikseiten hin- und herzuschalten.

Wenn das Bild fertig gezeichnet ist, kommt man mit der Leertaste wieder in den Textbildschirm. Dort wird man gefragt, ob noch etwas an der Grafik zu ändern ist. Wenn diese Frage bejaht wird, gelangt man in einen kleinen Grafikeditor. Man bewegt das Kreuz mit dem Joystick (Port 2) über den Bildschirm. Bei gedrücktem Feuerknopf werden Punkte gesetzt oder gelöscht. Mit den Tasten »%« und »L« kann man zwischen Punkt setzen und Punkt löschen hin- und herschalten. Verlassen kann man den Editor mit »E«.

5. Hardcopy

Dieser Menüpunkt sendet eine Hardcopy für einen Epson FX/RX-80 auf den seriellen Bus. Falls Sie über einen solchen Drucker verfügen, brauchen Sie nur noch die Sekundäradressen an Ihr Interface anpassen. In Zeile 395 setzen Sie die Sekundäradresse für Groß-/Klein-Schrift ein, in Zeile 407 die Sekundäradresse für den Epson-Direkt-Modus (ohne Auto-Linefeed !).

Sonstige Hinweise

— Falls Sie das Programm mit RUN/STOP unterbrochen haben und die Variablen noch vorhanden sind, können sie es mit »GOTO 115« wieder starten.
— Die Funktionstasten sind folgendermaßen belegt:
F1: Grafik 1
F3: Grafik 2
F5: Textbildschirm
F7: Textbildschirm
— Drücken Sie die Funktionstasten **nie** während Operationen auf dem seriellen Bus (Laden/Speichern/Hardcopy).
— Der Bereich von $9000 bis $9FFF ist aus Hardwaregründen unbenutzt (»Geisterbilder« des Zeichen-ROMs). Dort können eigene Hardcopyroutinen untergebracht werden.
— Man kann sich unter dem Menüpunkt »Bild berechnen« auch die Werte ausgeben lassen, um dann ein Bild von Hand zu zeichnen.

In Bild 3 bis Bild 6 sehen Sie noch einige Beispiele.

Zum Schluß möchte ich Ihnen noch viel Spaß bei der Arbeit mit meinem Programm wünschen.

(Oliver Günter/gk)

$0000	Zeropage
$0400	Textbildschirm
$0800	Programm und Variablen
$8000	Farbspeicher 2
$9000	frei
$A000	Basic-ROM und Grafik 2
$C000	Farbspeicher 1
$C400	Grafikerweiterung
$D000	I/O-Bereich
$E000	Kernal-ROM und Grafik 1

Tabelle 1: Speicherbelegung

SYS 50176	Grafik 1 ein
SYS 50179	Grafik 2 ein
SYS 50182	Textbildschirm ein
SYS 50185	Grafik 1 löschen
SYS 50188	Grafik 2 löschen
SYS 50191,F1,F2	Farbe setzen für Grafik 1+2
SYS 50194,X,Y	Punkt setzen
SYS 50197,X,Y	Punkt löschen
SYS 50200,X1,Y1,X2,Y2	Linie ziehen
SYS 50203,X1,Y1,X2,Y2	Linie löschen
SYS 50206,X,Y	Hinterschneidungsroutine
SYS 50209	Directory
SYS 50212	Funktionstasten initialisieren
SYS 50215,LF	Hardcopy

Tabelle 2: Befehle der Grafikerweiterung

Arrays:	
DX,DY,DZ	Eckpunktkoordinaten eines Körpers
WX,WY,WZ	Eckpunktkoordinaten nach dem Drehen
BX,BY	Bildschirmkoordinaten der Eckpunkte
MX,MY,MZ	Koordinaten der Flächenmittelpunkte
FL	Eckpunkte einer Fläche
PF	Punktanzahl einer Fläche
RF	Reihenfolge der Flächen
Strings:	
IN$	Gedrückte Taste
FU$	Funktionsstring (Überschrift)
FL$	Filename
Einfache Variablen:	
V	VIC-Basisadresse
Q,W,E,R	Laufvariablen
AP	Anzahl Punkte
AF	Anzahl Flächen
FX,FY,FZ	Koordinaten des Augpunktes
ZX,ZY,ZZ	Zwischenwertspeicher
AE	Vergrößerungsfaktor
JO	Joystickregister Port 2
XD,YD,ZD	Drehwinkel um X-, Y- und Z-Achse
FE,T,S	Nummer des Diskfehlers, Track, Sector
XP,YP,WI,WK	Hilfsvariablen der Drehroutine
M,N,B,C	Hilfsvariablen der Fluchtpunktroutine
PX,NX,PY,NY	Hilfsvariablen bei der Größenberechnung
AX,AY,BX,BY	Hilfsvariablen bei der Größenberechnung
LO	Löschvariable

Tabelle 3: Variablenliste

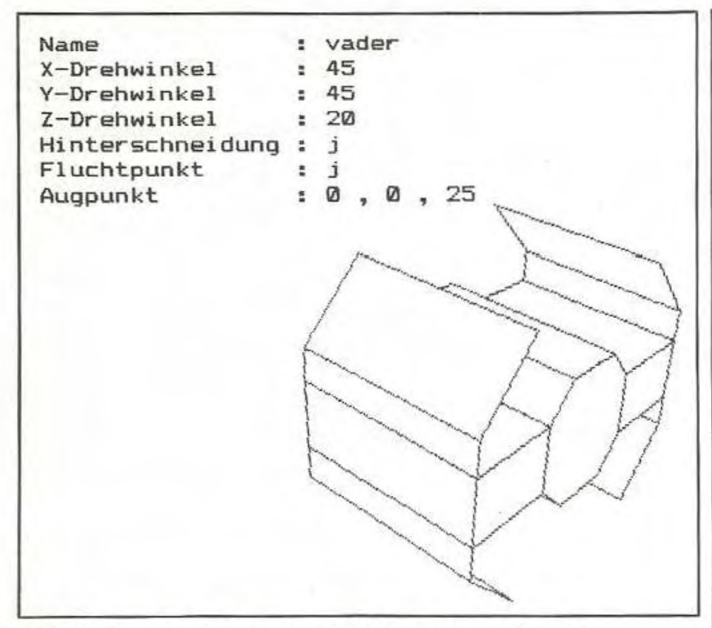


Bild 3. Die Hardcopy eines Bildes mit verdeckten »unsichtbaren« Linien

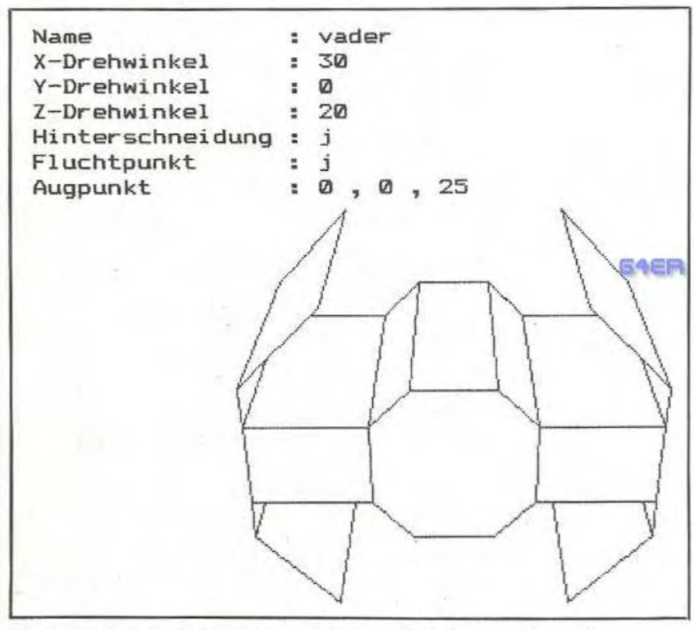


Bild 4. Die gleiche Figur wie in Bild 3, nur aus einem anderen Blickwinkel

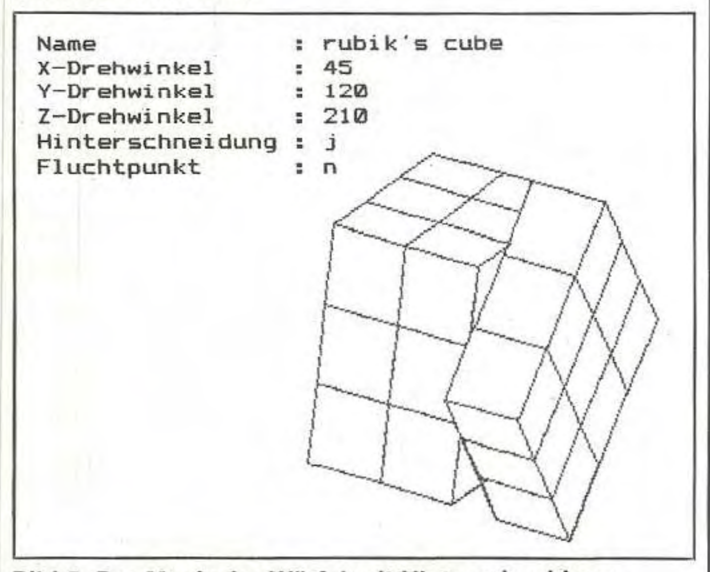


Bild 5. Der Magische Würfel mit Hinterschneidung

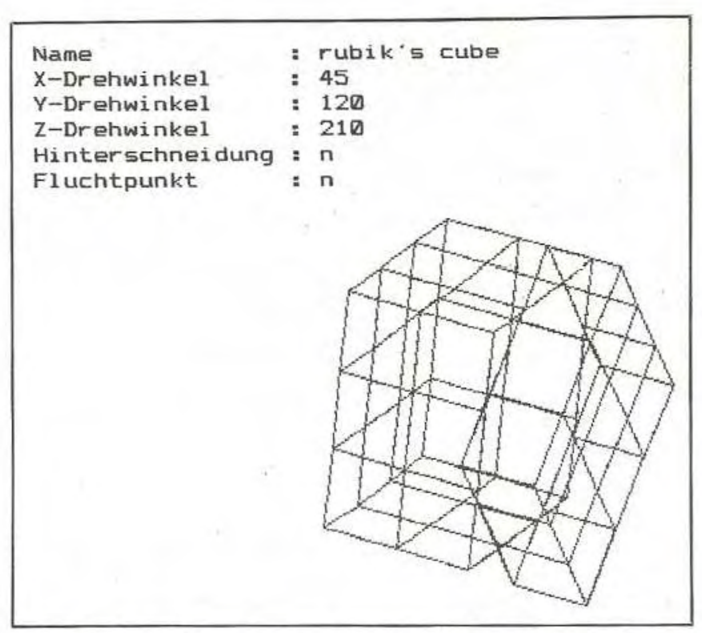


Bild 6. Bild 5 ohne Hinterschneidung. Er ist sozusagen durchsichtig.

```
100 IF C=1 THEN GOTO 115                              <017>
101 POKE 53280,15:POKE 53281,15:PRINT CHR$
    (147)CHR$(144)CHR$(9)CHR$(14)CHR$(8);             <140>
102 PRINT"FUNKTION : HAUPTPROGRAMM LADEN"             <003>
103 PRINT"{BLUE}*****************************
    ***************";                                 <210>
104 PRINT"*{10SPACE,BLACK}3D-SUPERGRAFIK{1
    4SPACE,BLUE}*";                                   <171>
105 PRINT"*{38SPACE}*";                               <236>
106 PRINT"*{10SPACE,BLACK}VON OLIVER GUENT
    ER{10SPACE,BLUE}*";                               <137>
107 PRINT"*********************************
    *******"                                          <245>
108 POKE 55,255:POKE 56,139                           <039>
109 FOR Q=0 TO 62                                     <164>
110 READ W                                            <070>
111 POKE 53184+Q,W                                    <173>
112 NEXT                                              <122>
113 C=1                                               <038>
114 LOAD"3D-1",8,1                                    <061>
115 PRINT"{GREY 3,3DOWN}LOAD";CHR$(34);"3D
    -2";CHR$(34);",8,1                                <120>
116 PRINT"{4DOWN}RUN"                                 <048>
117 PRINT"{8UP}";                                     <114>
118 FOR Q=0 TO 2:POKE 631+Q,13:NEXT                   <184>
119 POKE 198,3                                        <156>
120 DATA 0,0,0,0,0,0,0,16,0,0,16,0                    <217>
121 DATA 0,16,0,0,16,0,0,16,0,0,16,0                  <204>
122 DATA 0,0,0,0,0,0,31,131,240,0,0,0                 <126>
123 DATA 0,0,0,0,16,0,0,16,0,0,16,0                   <118>
124 DATA 0,16,0,0,16,0,0,16,0,0,0,0                   <182>
125 DATA 0,0,0                                        <247>
```


Listing 1. Das Ladeprogramm für 3D-Supergrafik

```
1 REM                                                 <063>
2 REM        3D-SUPERGRAFIK                           <106>
3 REM        BY OLIVER GUENTER                        <142>
4 REM        GRAEVINGHOLZSTR. 44                      <217>
5 REM        4600 DORTMUND 16                         <038>
6 REM        TEL: 0231/853317                         <107>
7 REM                                                 <069>
8 REM        FUER C-64 MIT VC-1541                    <232>
9 REM                                                 <071>
100 REM ********************************              <238>
101 REM *   INITIALISIERUNG            *              <074>
102 REM ********************************              <240>
103 SYS 50212:SYS 50182:SYS 50191,15,15               <110>
104 V=53248:POKE 50168,63                             <158>
105 POKE V+21,0:POKE V+32,15:POKE V+33,15             <241>
106 DIM DX(150),DY(150),DZ(150)                       <080>
107 DIM MX(150),MY(150),MZ(150)                       <041>
108 DIM BX(150),BY(150)                               <078>
109 DIM WX(120),WY(120),WZ(120)                       <152>
110 DIM FL(120,8)                                     <153>
111 DIM PF(120),RF(120)                               <237>
```

Listing 2. »3D-Supergrafik«

```
112 REM ********************************   <250>
113 REM *     HAUPTMENUE               *   <037>
114 REM ********************************   <252>
115 FU$="HAUPTMENUE":GOSUB 124             <132>
116 PRINT TAB(11);"{DOWN}-1- DATEN EINGEBE
    N"                                     <099>
117 PRINT TAB(11);"{DOWN}-2- DATEN LADEN"  <049>
118 PRINT TAB(11);"{DOWN}-3- DATEN SPEICHE
    RN"                                    <226>
119 PRINT TAB(11);"{DOWN}-4- BILD BERECHNE
    N"                                     <034>
120 PRINT TAB(11);"{DOWN}-5- HARDCOPY"     <130>
121 GET IN$:IN=VAL(IN$)                    <232>
122 ON IN GOTO 134,175,201,228,387         <146>
123 GOTO 121                               <115>
124 REM ********************************   <006>
125 REM *    TITEL + FUNKTION          *   <047>
126 REM ********************************   <008>
127 PRINT"{CLR,BLACK}FUNKTION : ";FU$      <180>
128 PRINT"{BLUE}**************************
    **************";                       <235>
129 PRINT"*{10SPACE,BLACK}3D-SUPERGRAFIK{1
    4SPACE,BLUE}*";                        <196>
130 PRINT"*{38SPACE}*";                    <005>
131 PRINT"*{10SPACE,BLACK}(C) OLIVER GUENT
    ER{10SPACE,BLUE}*";                    <044>
132 PRINT"********************************
    ********{BLACK}"                       <057>
133 RETURN                                 <191>
134 REM ********************************   <016>
135 REM *     DATEN EINGEBEN           *   <152>
136 REM ********************************   <018>
137 FU$="DATEN EINGEBEN":GOSUB 124         <063>
138 INPUT"{DOWN}WIE VIELE PUNKTE{3SPACE}";
    AP                                     <001>
139 IF AP=0 THEN GOTO 112                  <061>
140 INPUT"{DOWN}WIE VIELE FLAECHEN ";AF    <226>
141 IF AF=0 THEN GOTO 112                  <158>
142 FOR Q=1 TO AP                          <136>
143 FU$="PUNKTE EINGEBEN":GOSUB 124        <244>
144 PRINT"{DOWN}PUNKT :";Q                 <119>
145 INPUT"{2DOWN}X-KOORDINATE ";DX(Q)      <248>
146 INPUT"{DOWN}Y-KOORDINATE ";DY(Q)       <125>
147 INPUT"{DOWN}Z-KOORDINATE ";DZ(Q)       <174>
148 PRINT"{DOWN}RICHTIG ?"                 <223>
149 POKE 198,0:WAIT 198,1                  <121>
150 IF PEEK(631)=78 THEN PRINT"{10UP}":POK
    E 198,0:GOTO 143                       <126>
151 NEXT                                   <161>
152 POKE 198,0                             <060>
153 FOR Q=1 TO AF                          <137>
154 FU$="FLAECHEN EINGEBEN":GOSUB 124      <199>
155 PRINT"{DOWN}FLAECHE :";Q;"{DOWN}"      <245>
156 R=0                                    <125>
157 FOR W=1 TO 8                           <243>
158 IF R=1 THEN NEXT:GOTO 168              <030>
159 PRINT"PUNKT :";W;                      <010>
160 T$=""                                  <009>
161 INPUT T$                               <155>
162 T=VAL(T$)                              <057>
163 IF T<0 OR T>AP OR INT(T)<>T THEN PRINT
    "{2UP}":GOTO 159                       <120>
164 IF RIGHT$(T$,1)="L"THEN T=-T           <030>
165 FL(Q,W)=T                              <097>
166 IF T=0 THEN R=1:PF(Q)=W-1:NEXT         <018>
167 NEXT:PF(Q)=8                           <111>
168 PRINT"{DOWN}RICHTIG ?"                 <243>
169 POKE 198,0:WAIT 198,1                  <141>
170 IF PEEK(631)=78 THEN PRINT"{HOME,9DOWN
    }":POKE 198,0:GOTO 154                 <004>
171 NEXT                                   <181>
172 PRINT"{2DOWN,RVSON}< SPACE >{RVOFF}"   <105>
173 GET IN$:IF IN$<>" "THEN 173            <111>
174 GOTO 112                               <182>
175 REM ********************************   <057>
176 REM *     DATEN LADEN              *   <022>
177 REM ********************************   <059>
178 FU$="DATEN LADEN ($=DIR)":GOSUB 124    <184>
179 INPUT"{DOWN}FILENAME ";FL$             <168>
180 IF FL$="@"THEN GOTO 112                <076>
181 IF FL$="$"THEN PRINT:SYS 50209:GOTO 17
    9                                      <053>
182 OPEN 15,8,15:OPEN 1,8,2,"3D-"+FL$+",S,
    R"                                     <079>
183 GOSUB 423                              <023>
184 IF ER=1 THEN CLOSE 1:CLOSE 15:GOTO 179 <039>
185 GET#1,A$:AP=ASC(A$)                    <224>
186 FOR Q=1 TO AP                          <180>
187 INPUT#1,A$:DX(Q)=VAL(A$)               <030>
188 INPUT#1,A$:DY(Q)=VAL(A$)               <033>
189 INPUT#1,A$:DZ(Q)=VAL(A$)               <036>
190 NEXT                                   <200>
191 GET#1,A$:AF=ASC(A$)                    <190>
192 FOR Q=1 TO AF                          <176>
193 GET#1,A$:PF(Q)=ASC(A$)                 <090>
194 FOR W=1 TO PF(Q)                       <072>
195 GET#1,A$:FL(Q,W)=ASC(A$)-128           <011>
196 NEXT                                   <206>
197 NEXT                                   <207>
198 CLOSE 1                                <209>
199 CLOSE 15                               <037>
200 GOTO 112                               <208>
201 REM ********************************   <083>
202 REM *    DATEN SPEICHERN           *   <033>
203 REM ********************************   <085>
204 FU$="DATEN SPEICHERN ($=DIR)":GOSUB 12
    4                                      <165>
205 GOSUB 413:IF ER=1 THEN GOTO 112        <109>
206 INPUT"{DOWN}FILENAME ";FL$             <195>
207 IF FL$="@"THEN GOTO 112                <103>
208 IF FL$="$"THEN PRINT:SYS 50209:GOTO 20
    6                                      <046>
209 OPEN 15,8,15:OPEN 1,8,2,"3D-"+FL$+",S,
    W"                                     <111>
210 GOSUB 423                              <050>
211 IF ER=1 THEN CLOSE 1:CLOSE 15:GOTO 206 <041>
212 PRINT#1,CHR$(AP);                      <177>
213 FOR Q=1 TO AP                          <207>
214 PRINT#1,DX(Q)                          <216>
215 PRINT#1,DY(Q)                          <025>
216 PRINT#1,DZ(Q)                          <090>
217 NEXT                                   <227>
218 PRINT#1,CHR$(AF);                      <173>
219 FOR Q=1 TO AF                          <203>
220 PRINT#1,CHR$(PF(Q));                   <208>
221 FOR W=1 TO PF(Q)                       <099>
222 PRINT#1,CHR$(FL(Q,W)+128);             <237>
223 NEXT                                   <233>
224 NEXT                                   <234>
225 CLOSE 1                                <236>
226 CLOSE 15                               <064>
227 GOTO 112                               <235>
228 REM ********************************   <110>
229 REM *     BILD BERECHNEN           *   <073>
230 REM ********************************   <112>
231 FU$="BILD BERECHNEN":GOSUB 124         <040>
232 GOSUB 413:IF ER=1 THEN GOTO 112        <136>
233 INPUT"{DOWN}X-DREHWINKEL ";XD          <243>
234 INPUT"{DOWN}Y-DREHWINKEL ";YD          <028>
235 INPUT"{DOWN}Z-DREHWINKEL ";ZD          <069>
236 INPUT"{DOWN}HINTERSCHNEIDUNG (J/N) ";H
    I$                                     <218>
237 IF HI$="J"THEN HI=1:GOTO 240           <049>
238 IF HI$<>"N"THEN PRINT"{3UP}":GOTO 236  <229>
239 HI=0                                   <139>
240 INPUT"{DOWN}FLUCHTPUNKT (J/N) ";F$     <082>
241 IF F$="J"THEN F=1:GOTO 245             <040>
242 IF F$<>"N"THEN PRINT"{3UP}":GOTO 240   <153>
243 F=0                                    <164>
244 GOTO 247                               <212>
245 INPUT"{DOWN}AUGPUNKT-KOOR.{3SPACE}0,0,
    25{8LEFT}";FX,FY,FZ                    <127>
246 IF FZ<=0 THEN PRINT"{3UP}":GOTO 245    <121>
247 INPUT"{DOWN}WERTE AUSGEBEN (J/N){3SPAC
    E}N{3LEFT}";WE$                        <029>
248 IF WE$="J"THEN WE=1:GOTO 251           <184>
249 IF WE$<>"N"THEN PRINT"{3UP}":GOTO 247  <136>
250 WE=0                                   <178>
251 PRINT"{2DOWN}DREHE PUNKTE"             <018>
252 FOR Q=1 TO AP                          <246>
253 WX(Q)=DX(Q)                            <027>
254 WY(Q)=DY(Q)                            <038>
255 WZ(Q)=DZ(Q)                            <049>
256 NEXT                                   <012>
257 IF XD=0 THEN GOTO 262                  <194>
258 FOR Q=1 TO AP                          <254>
259 WI=XD:XP=WY(Q):YP=WZ(Q):GOSUB 434      <111>
260 WY(Q)=XP:WZ(Q)=YP                      <077>
261 NEXT                                   <017>
262 IF YD=0 THEN GOTO 267                  <247>
263 FOR Q=1 TO AP                          <003>
264 WI=YD:XP=WZ(Q):YP=WX(Q):GOSUB 434      <100>
265 WZ(Q)=XP:WX(Q)=YP                      <058>
266 NEXT                                   <022>
267 IF YD=0 THEN GOTO 272                  <216>
```

Listing 2. Hauptprogramm »3D-Supergrafik«. Bitte mit dem Checksummer 64 eingeben.

```
268 FOR Q=1 TO AP                                  <008>
269 WI=ZD:XP=WX(Q):YP=WY(Q):GOSUB 434              <137>
270 WX(Q)=XP:WY(Q)=YP                              <063>
271 NEXT                                           <027>
272 IF F=1 THEN GOTO 278                           <240>
273 FOR Q=1 TO AP                                  <013>
274 BX(Q)=WX(Q)                                    <241>
275 BY(Q)=WY(Q)                                    <252>
276 NEXT                                           <032>
277 GOTO 285                                       <247>
278 PRINT"{UP}BERECHNE BILDPUNKTE"                 <071>
279 FOR Q=1 TO AP                                  <019>
280 X1=FX:Y1=FY:Z1=FZ                              <146>
281 X2=WX(Q):Y2=WY(Q):Z2=WZ(Q)                     <084>
282 GOSUB 446                                      <252>
283 BX(Q)=X1:BY(Q)=Y1                              <149>
284 NEXT                                           <040>
285 IF HI=0 THEN GOTO 298                          <234>
286 PRINT"{UP}BERECHNE MITTELPUNKTE"               <157>
287 FOR Q=1 TO AF                                  <017>
288 ZX=0:ZY=0:ZZ=0                                 <010>
289 FOR W=1 TO PF(Q)                               <169>
290 ZX=ZX+BX(ABS(FL(Q,W)))                         <238>
291 ZY=ZY+BY(ABS(FL(Q,W)))                         <057>
292 ZZ=ZZ+WZ(ABS(FL(Q,W)))                         <153>
293 NEXT                                           <049>
294 MX(Q)=ZX/PF(Q)                                 <001>
295 MY(Q)=ZY/PF(Q)                                 <012>
296 MZ(Q)=ZZ/PF(Q)                                 <023>
297 NEXT                                           <053>
298 PRINT"{UP}BERECHNE ACHSEINHEITEN"              <241>
299 PX=0:NX=0:PY=0:NY=0                            <043>
300 FOR Q=1 TO AP                                  <040>
301 ZX=BX(Q)                                       <154>
302 ZY=BY(Q)                                       <227>
303 GOSUB 458                                      <098>
304 NEXT                                           <060>
305 AX=-(PX+NX)/2                                  <071>
306 AY=-(PY+NY)/2                                  <089>
307 BX=(ABS(PX)+ABS(NX))/3.2                       <011>
308 BY=(ABS(PY)+ABS(NY))/2                         <102>
309 AE=BX                                          <239>
310 IF BX<BY THEN AE=BY                            <209>
311 AE=AE*100/98                                   <186>
312 FOR Q=1 TO AF                                  <042>
313 RF(Q)=Q                                        <086>
314 NEXT                                           <070>
315 IF HI=0 THEN GOTO 325                          <214>
316 PRINT"{UP}BERECHNE REIHENFOLGE{2SPACE}
    "                                              <240>
317 FOR Q=1 TO AF-1                                <074>
318 FOR W=Q+1 TO AF                                <214>
319 IF MZ(RF(Q))<=MZ(RF(W))THEN GOTO 323           <047>
320 A=RF(Q)                                        <164>
321 RF(Q)=RF(W)                                    <215>
322 RF(W)=A                                        <015>
323 NEXT                                           <079>
324 NEXT                                           <080>
325 PRINT"{UP}ZEICHNE BILD{10SPACE}"               <009>
326 SYS 50185:SYS 50176                            <142>
327 FOR Q=1 TO AF                                  <057>
328 SYS 50188                                      <192>
329 NF=RF(Q)                                       <115>
330 FOR W=1 TO PF(NF)                              <065>
331 ZX=BX(ABS(FL(NF,W)))                           <030>
332 ZY=BY(ABS(FL(NF,W)))                           <103>
333 GOSUB 466                                      <079>
334 E=W+1:IF E>PF(NF)THEN E=1                      <222>
335 X=ZX:Y=ZY                                      <252>
336 ZX=BX(ABS(FL(NF,E)))                           <017>
337 ZY=BY(ABS(FL(NF,E)))                           <090>
338 GOSUB 466                                      <084>
339 R=SGN(FL(NF,E))                                <059>
340 LO=0                                           <050>
341 IF R=1 THEN GOTO 344                           <128>
342 IF HI=0 THEN GOTO 347                          <009>
343 LO=3                                           <149>
344 SYS 50200+LO,X,Y,ZX,ZY                         <097>
345 IF LO=3 THEN SYS 50194,X,Y                     <083>
346 IF LO=3 THEN SYS 50194,ZX,ZY                   <090>
347 NEXT                                           <103>
348 IF HI=0 THEN GOTO 353                          <243>
349 ZX=MX(NF)                                      <006>
350 ZY=MY(NF)                                      <079>
351 GOSUB 466                                      <097>
352 SYS 50206,ZX,ZY                                <103>
353 NEXT                                           <109>
354 PRINT"{UP,RVSON}< SPACE >{RVOFF,3SPACE
    }"                                             <123>
355 GET IN$:IF IN$<>" "THEN GOTO 355               <040>
356 SYS 50182                                      <217>
357 INPUT"{UP}BILD AENDERN (J/N) ";AE$             <038>
358 IF AE$="N"THEN GOTO 376                        <087>
359 IF AE$<>"J"THEN GOTO 357                       <158>
360 X=160:Y=100:LO=3:SYS 50176:POKE V+21,1         <218>
361 GET IN$:IF IN$="S"THEN LO=0                    <028>
362 IF IN$="L"THEN LO=3                           <175>
363 IF IN$="E"THEN POKE V+1,0:GOTO 373             <069>
364 JO=PEEK(249)                                   <001>
365 IF(JO AND 1)=0 AND Y>0 THEN Y=Y-1              <246>
366 IF(JO AND 2)=0 AND Y<199 THEN Y=Y+1            <230>
367 IF(JO AND 4)=0 AND X>0 THEN X=X-1              <101>
368 IF(JO AND 8)=0 AND X<319 THEN X=X+1            <067>
369 IF(JO AND 16)=0 THEN SYS 50194+LO,X,Y          <001>
370 POKE V,((X+13)AND 255):POKE V+1,Y+40           <207>
371 POKE V+16,((X+13)AND 256)/256                  <005>
372 GOTO 361                                       <190>
373 PRINT"{UP,RVSON}< SPACE >{RVOFF,13SPAC
    E}"                                            <142>
374 SYS 50182                                      <235>
375 GET IN$:IF IN$<>" "THEN GOTO 375               <068>
376 IF WE=0 THEN GOTO 112                          <043>
377 FOR Q=1 TO AP                                  <117>
378 FU$="WERTE AUSGEBEN":GOSUB 124                 <049>
379 PRINT"{DOWN}PUNKT :";Q                         <100>
380 PRINT"{2DOWN}X-KOORDINATE :";BX(Q)             <107>
381 PRINT"{DOWN}Y-KOORDINATE :";BY(Q)              <229>
382 IF F=0 THEN PRINT"{DOWN}Z-KOORDINATE :
    ";WZ(Q)                                        <241>
383 PRINT"{2DOWN,RVSON}< SPACE{SHIFT-SPACE
    }>{RVOFF}"                                     <208>
384 GET IN$:IF IN$<>" "THEN GOTO 384               <073>
385 NEXT                                           <141>
386 GOTO 112                                       <140>
387 REM ********************************          <015>
388 REM *   HARDCOPY EPSON RX-80         *          <011>
389 REM ********************************          <017>
390 FU$="HARDCOPY":GOSUB 124                       <049>
391 INPUT"{DOWN}IST DER DRUCKER WIRKLICH A
    N ";D$                                         <068>
392 IF D$="J"THEN GOTO 395                         <136>
393 IF D$<>"N"THEN PRINT"{3UP}":GOTO 391           <154>
394 GOTO 112                                       <148>
395 OPEN 4,4,2:REM ++ GROSS/KLEIN ++               <192>
396 PRINT#4                                        <145>
397 PRINT#4,"NAME{13SPACE}: ";FL$                  <091>
398 PRINT#4,"X-DREHWINKEL{5SPACE}:";XD             <159>
399 PRINT#4,"Y-DREHWINKEL{5SPACE}:";YD             <000>
400 PRINT#4,"Z-DREHWINKEL{5SPACE}:";ZD             <097>
401 PRINT#4,"HINTERSCHNEIDUNG : ";HI$              <243>
402 PRINT#4,"FLUCHTPUNKT{6SPACE}: ";F$             <010>
403 IF F=1 THEN PRINT#4,"AUGPUNKT{9SPACE}:
    ";FX;",";FY;",";FZ                             <028>
404 PRINT#4                                        <153>
405 PRINT#4                                        <154>
406 CLOSE 4                                        <187>
407 OPEN 4,4,1:REM ++ EPSON MODE ++                <164>
408 SYS 50215,4                                    <128>
409 PRINT#4,CHR$(10)                               <124>
410 PRINT#4,CHR$(10)                               <125>
411 CLOSE 4                                        <192>
412 GOTO 112                                       <166>
413 REM ********************************          <041>
414 REM *    AUF DATEN TESTEN            *          <132>
415 REM ********************************          <043>
416 ER=0                                           <122>
417 IF AP>0 THEN RETURN                            <131>
418 PRINT"{DOWN,RVSON}KEINE DATEN VORHANDE
    N !{RVOFF}"                                    <050>
419 PRINT"{2DOWN,RVSON}< SPACE >{RVOFF}"           <098>
420 GET IN$:IF IN$<>" "THEN GOTO 420               <055>
421 ER=1                                           <159>
422 RETURN                                         <226>
423 REM ********************************          <051>
424 REM *   FEHLERKANAL ABFRAGEN         *          <065>
425 REM ********************************          <053>
426 ER=0                                           <132>
427 INPUT#15,FE,FE$,T,S                            <065>
428 IF FE=0 THEN RETURN                            <037>
429 PRINT"{DOWN,RVSON}DISKFEHLER :{RVOFF,S
    PACE}";FE$                                     <214>
430 PRINT"{DOWN,RVSON}TRACK{6SPACE}:{RVOFF
    }";T                                           <002>
431 PRINT"{DOWN,RVSON}SEKTOR{5SPACE}:{RVOF
    F}";S                                          <064>
```

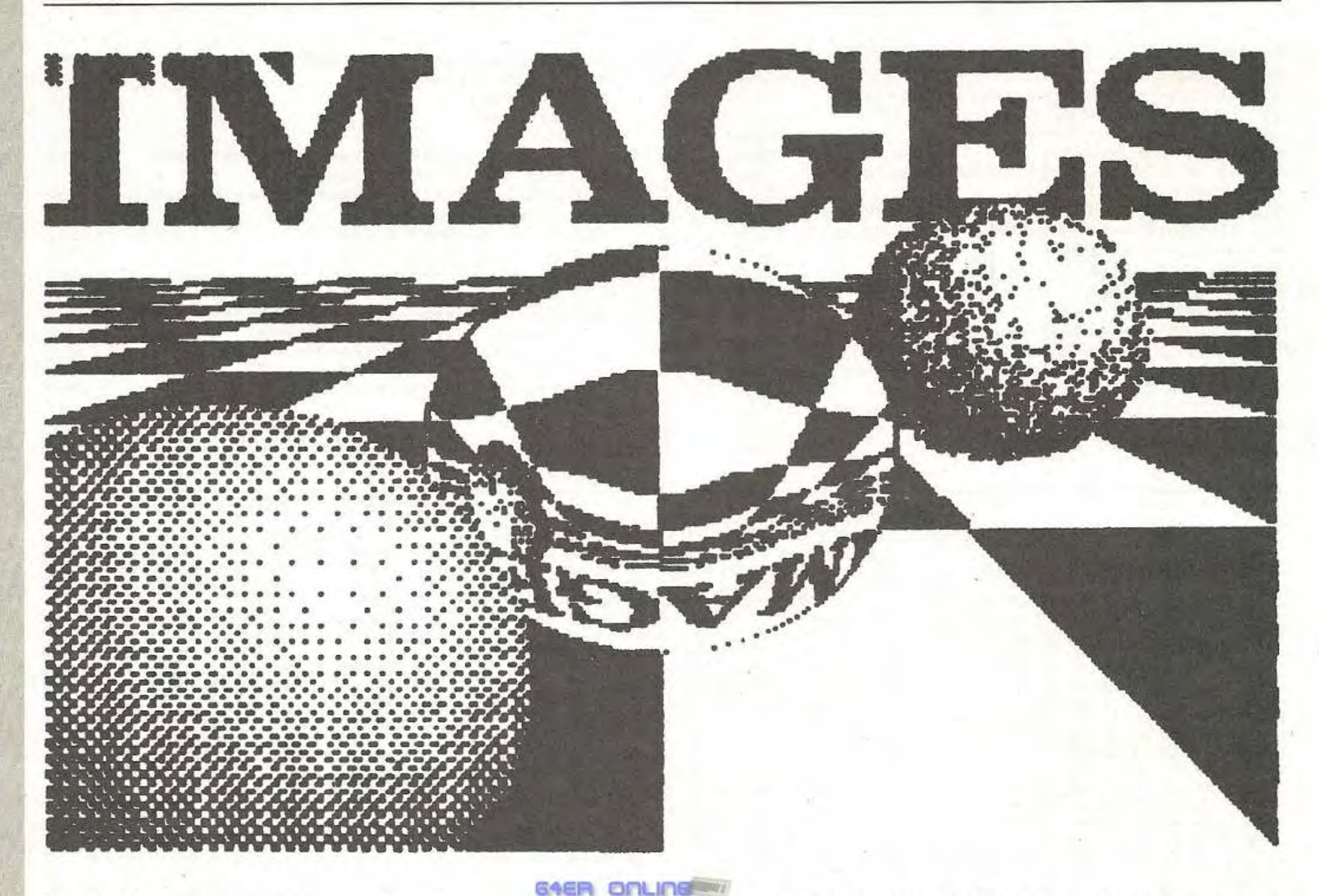
Listing 2. Hauptprogramm »3D-Supergrafik« (Fortsetzung)

```
432 ER=1                                         <170>
433 RETURN                                       <237>
434 REM ******************************           <062>
435 REM *   REC->POL + WI POL->REC      *        <072>
436 REM ******************************           <064>
437 LA=SQR(XP↑2+YP↑2)                            <000>
438 IF XP=0 THEN WK=π/2*SGN(YP):GOTO 442         <206>
439 WK=ATN(YP/XP)                                <169>
440 IF SGN(XP)=-1 THEN WK=WK+π*SGN(YP)           <252>
441 IF SGN(XP)=-1 AND SGN(YP)=0 THEN WK=π        <239>
442 WK=WK+(WI*π/180)                             <173>
443 XP=COS(WK)*LA                                <028>
444 YP=SIN(WK)*LA                                <065>
445 RETURN                                       <249>
446 REM ******************************           <074>
447 REM *   SCHNITTPT. GERADE/EBENE     *        <194>
448 REM ******************************           <076>
449 IF X2=X1 THEN GOTO 453                       <064>
450 M=(Z2-Z1)/(X2-X1)                            <213>
451 B=Z1-M*X1                                    <222>
452 X1=-B/M                                      <003>
453 IF Y2=Y1 THEN GOTO 457                       <204>
454 N=(Z2-Z1)/(Y2-Y1)                            <254>
455 C=Z1+N*Y1                                    <041>
456 Y1=-C/N                                      <076>
457 RETURN                                       <005>
458 REM ******************************           <086>
459 REM *   AE BERECHNEN                *        <169>
460 REM ******************************           <088>
461 IF PX<ZX THEN PX=ZX                          <041>
462 IF NX>ZX THEN NX=ZX                          <214>
463 IF PY<ZY THEN PY=ZY                          <223>
464 IF NY>ZY THEN NY=ZY                          <140>
465 RETURN                                       <013>
466 REM ******************************           <094>
467 REM *   BILDKOOR. BERECHNEN         *        <071>
468 REM ******************************           <096>
469 ZX=160+((ZX+AX)*100/AE)                      <084>
470 ZY=100-((ZY+AY)*100/AE)                      <109>
471 RETURN                                       <019>
```


Listing 2. »3D-Supergrafik« (Schluß)

```
 programm : 3d-1              c400 caae
------------------------------------
c400 : 4c 2a c4 4c 33 c4 4c 3c  1f
c408 : c4 4c 4c c4 4c 5b c4 4c  e9
c410 : 6a c4 4c 9c c4 4c b9 c4  a2
c418 : 4c 8d c5 4c 94 c5 4c 47  5d
c420 : c7 4c 4d c9 4c b8 c9 4c  e4
c428 : 15 ca a9 3b a2 08 a0 00  61
c430 : 4c 42 c4 a9 3b a2 38 a0  ef
c438 : 01 4c 42 c4 a9 1b a2 17  b5
c440 : a0 03 8d 11 d0 8e 18 d0  6b
c448 : 8c 00 dd 60 a0 00 84 fa  6a
c450 : a9 e0 85 fb a9 00 a2 20  af
c458 : 4c 8f c4 a0 00 84 fa a9  14
c460 : a0 85 fb a9 00 a2 20 4c  25
c468 : 8f c4 a0 00 84 fa a9 c0  ca
c470 : 85 fb 20 fd ae 20 9e b7  90
c478 : 8a a2 04 20 8f c4 a0 00  fa
c480 : 84 fa a9 8c 85 fb 20 fd  32
c488 : ae 20 9e b7 8a a2 04 a0  f4
c490 : 00 91 fa c8 d0 fb e6 fb  b1
c498 : ca d0 f4 60 20 d6 c4 20  20
c4a0 : ff c4 a9 00 85 fe a9 e0  24
c4a8 : 85 ff 20 5d c5 a9 00 85  95
c4b0 : fe a9 a0 85 ff 20 5d c5  5e
c4b8 : 60 20 d6 c4 20 ff c4 a9  df
c4c0 : 00 85 fe a9 e0 85 ff 20  f2
c4c8 : 62 c5 a9 00 85 fe a9 a0  af
c4d0 : 85 ff 20 62 c5 60 20 fd  85
c4d8 : ae 20 eb b7 86 02 a5 14  bf
c4e0 : 85 fa a5 15 85 fb e0 c8  3c
c4e8 : 90 06 20 3c c4 4c 48 b2  40
c4f0 : a5 fb c9 01 90 08 d0 f2  98
c4f8 : a5 fa c9 40 b0 ec 60 a5  d4
c500 : 02 29 f8 85 fe 85 fc a9  e9
c508 : 00 85 fd 06 fc 26 fd 06  10
c510 : fc 26 fd 18 a5 fc 65 fe  78
c518 : 85 fc a5 fd 69 00 85 fd  ed
c520 : 06 fc 26 fd 06 fc 26 fd  ca
c528 : 06 fc 26 fd a5 02 29 07  13
c530 : 18 65 fc 85 fc a5 fd 69  b2
c538 : 00 85 fd 18 a5 fa 29 f8  46
c540 : 65 fc 85 fc a5 fb 65 fd  f0
c548 : 85 fd 18 a5 fa 29 07 49  2e
c550 : 07 aa a9 01 ca 30 03 0a  85
c558 : d0 fa 85 02 60 38 08 4c  c7
c560 : 64 c5 18 08 18 a5 fe 65  23
c568 : fc 85 fa a5 ff 65 fd 85  c8
c570 : fb a5 02 a0 00 a2 34 28  09
c578 : 78 86 01 90 04 11 fa b0  9c
c580 : 04 49 ff 31 fa 91 fa a2  bc
c588 : 37 86 01 58 60 a9 ff 85  ac
c590 : 97 4c 98 c5 a9 00 85 97  0c
c598 : 20 d6 c4 a5 02 8d ae ca  e6
c5a0 : a5 fa 8d b7 ca a5 fb 8d  01
c5a8 : b8 ca 20 d6 c4 a5 02 8d  45
c5b0 : af ca a5 fa 8d b0 ca ad  72
c5b8 : b7 ca 85 14 a5 fb 8d b1  8c
c5c0 : ca ad b8 ca 85 15 a0 01  6e
c5c8 : 8c bc ca 8c b9 ca 8c b7  8a
c5d0 : ca 88 8c b8 ca 8c bb ca  ae
c5d8 : 8c ba ca 88 ad b1 ca c5  a4
c5e0 : 15 90 09 d0 1e ad b0 ca  41
c5e8 : c5 14 b0 17 38 a5 14 ed  a3
c5f0 : b0 ca 8d b5 ca a5 15 ed  29
c5f8 : b1 ca 8d b6 ca 8c bc ca  e2
c600 : 4c 14 c6 38 ad b0 ca e5  66
c608 : 14 8d b5 ca ad b1 ca e5  09
c610 : 15 8d b6 ca ad af ca cd  12
c618 : ae ca b0 10 38 ad ae ca  9b
c620 : ed af ca 8d b4 ca 8c b9  90
c628 : ca 4c 32 c6 ed ae ca 8d  18
c630 : b4 ca ad b6 ca d0 24 ad  ab
c638 : b5 ca cd b4 ca b0 1c ae  5c
c640 : b4 ca 8d b4 ca 8e b5 ca  e1
c648 : ad bc ca 8d ba ca ad b9  e4
c650 : ca 8d bb ca c8 8c bc ca  a2
c658 : 8c b9 ca ad b6 ca 4a 8d  2f
c660 : b3 ca ad b5 ca 6a 8d b2  36
c668 : ca 4c ff c6 ad bc ca 30  7d
c670 : 0e 18 65 14 85 14 a5 15  20
c678 : 69 00 85 15 4c 8c c6 38  9a
c680 : a5 14 e9 01 85 14 a5 15  84
c688 : e9 00 85 15 18 ad ae ca  b5
c690 : 6d bb ca 8d ae ca 18 ad  3c
c698 : b2 ca 6d b4 ca 8d b2 ca  1b
c6a0 : ad b3 ca 69 00 8d b3 ca  d7
c6a8 : ee b7 ca d0 03 ee b8 ca  5f
c6b0 : ad b3 ca cd b6 ca 90 47  36
c6b8 : d0 08 ad b5 ca cd b2 ca  2a
c6c0 : b0 3d 38 ad b2 ca ed b5  77
c6c8 : ca 8d b2 ca ad b3 ca ed  de
c6d0 : b6 ca 8d b3 ca ad ba ca  60
c6d8 : 30 0e 18 65 14 85 14 a5  cb
c6e0 : 15 69 00 85 15 4c f5 c6  74
c6e8 : 38 a5 14 e9 01 85 14 a5  0d
c6f0 : 15 e9 00 85 15 18 ad ae  d1
c6f8 : ca 6d b9 ca 8d ae ca ad  15
c700 : ae ca 85 02 a5 14 85 fa  bc
c708 : a5 15 85 fb 20 23 c7 ad  ae
c710 : b8 ca cd b6 ca 90 09 ad  28
c718 : b5 ca cd b7 ca b0 01 60  94
c720 : 4c 6c c6 20 ff c4 a9 00  25
c728 : 85 fe a9 a0 85 ff 20 5d  3e
c730 : c5 a9 00 85 fe a9 e0 85  46
c738 : ff a5 97 f0 04 20 5d c5  50
c740 : 60 20 62 c5 60 a5 fa 20  61
c748 : d6 c4 a5 fa 8d bd ca a5  86
c750 : fb 8d be ca a5 02 8d bf  3b
c758 : ca a9 00 8d c6 ca 8d c7  31
c760 : ca 8d c8 ca a0 01 8c c3  48
c768 : ca 8d c4 ca 8c c5 ca 20  e6
c770 : 08 c8 ad bd ca 85 fa ad  1f
c778 : be ca 85 fb ad bf ca 85  8b
c780 : 02 20 00 c9 a5 02 f0 01  fc
c788 : 60 ad bd ca 85 fa ad be  ec
c790 : ca 85 fb 18 ad bf ca 6d  fe
c798 : c5 ca 85 02 20 00 c9 a5  d8
c7a0 : 02 f0 05 a9 01 8d c8 ca  c6
c7a8 : ad c3 ca 49 fe 8d c3 ca  14
c7b0 : ad c4 ca 49 ff 8d c4 ca  b0
c7b8 : 20 3c c8 ad c3 ca 49 fe  94
c7c0 : 8d c3 ca ad c4 ca 49 ff  5f
c7c8 : 8d c4 ca 20 3c c8 18 ad  34
c7d0 : bf ca 6d c5 ca 8d bf ca  b6
c7d8 : ad c8 ca d0 03 4c f5 c7  b0
c7e0 : ad c5 ca c9 01 f0 01 60  b8
c7e8 : a9 ff 8d c5 ca a9 00 8d  c2
c7f0 : c8 ca 20 22 c8 ad c7 ca  18
c7f8 : d0 03 4c 89 c7 a9 00 8d  73
c800 : c7 ca 20 22 c8 4c 89 c7  1d
c808 : ae c6 ca ad bd ca 9d c9  be
c810 : ca ad be ca 9d d3 ca ad  b9
c818 : bf ca 9d dd ca e8 8e c6  1b
c820 : ca 60 ae c6 ca ca bd c9  2c
c828 : ca 8d bd ca bd d3 ca 8d  42
c830 : be ca bd dd ca 8d bf ca  2c
c838 : 8e c6 ca 60 ad bd ca 8d  f7
c840 : c0 ca ad be ca 8d c1 ca  5e
c848 : ad bf ca 8d c2 ca ad bd  ee
c850 : ca 85 fa ad be ca 85 fb  a1
c858 : ad bf ca 85 02 20 e6 c8  96
c860 : 18 ad bd ca 6d c3 ca 85  43
c868 : fa ad be ca 6d c4 ca 85  75
c870 : fb ad bf ca 85 02 20 00  74
c878 : c9 a5 02 f0 13 ad c0 ca  ea
c880 : 8d bd ca ad c1 ca 8d be  7a
c888 : ca ad c2 ca 8d bf ca 60  f5
c890 : 18 ad bd ca 6d c3 ca 8d  83
c898 : bd ca 85 fa ad be ca 6d  52
c8a0 : c4 ca 8d be ca 85 fb ad  29
c8a8 : c8 ca d0 03 4c 4e c8 18  f4
c8b0 : ad bf ca 6d c5 ca 85 02  6a
c8b8 : 20 00 c9 a5 02 f0 03 4c  4c
c8c0 : 4e c8 18 ad bf ca 6d c5  c2
c8c8 : ca 8d bf ca 20 08 c8 38  78
c8d0 : ad bf ca ed c5 ca 8d bf  35
c8d8 : ca a9 00 8d c8 ca a9 01  b4
c8e0 : 8d c7 ca 4c 4e c8 20 ff  39
c8e8 : c4 a9 00 85 fe a9 e0 85  fd
c8f0 : ff 20 62 c5 a9 00 85 fe  ff
c8f8 : a9 a0 85 ff 20 5d c5 60  17
c900 : 20 12 c9 20 ff c4 a9 00  6c
c908 : 85 fe a9 a0 85 ff 20 2d  be
c910 : c9 60 a6 02 e0 c8 90 06  96
c918 : 20 3c c4 4c 48 b2 a5 fb  ba
c920 : c9 01 90 08 d0 f2 a5 fa  c0
c928 : c9 40 b0 ec 60 18 a5 fe  36
c930 : 65 fc 85 fa a5 ff 65 fd  c0
c938 : 85 fb a5 02 a0 00 a2 34  61
c940 : 78 86 01 31 fa 85 02 a2  8b
c948 : 37 86 01 58 60 a9 b0 85  2f
c950 : bb a9 c9 85 bc a9 08 85  47
c958 : b7 85 ba a9 60 85 b9 20  0f
c960 : d5 f3 a5 ba 20 b4 ff a5  e2
c968 : b9 20 96 ff a9 00 85 90  a9
c970 : a0 03 84 fb 20 a5 ff 85  6d
c978 : fc a4 90 d0 2f 20 a5 ff  8f
c980 : a4 90 d0 28 a4 fb 88 d0  93
c988 : e9 a6 fc 20 cd bd a9 20  b9
c990 : 20 d2 ff 20 a5 ff a6 90  33
c998 : d0 12 aa f0 06 20 d2 ff  e7
c9a0 : 4c 93 c9 a9 0d 20 d2 ff  7a
c9a8 : a0 02 d0 c6 20 42 f6 60  07
c9b0 : 24 3a 33 44 2d 2a 3d 53  06
c9b8 : 78 a9 c5 8d 14 03 a9 c9  bb
c9c0 : 8d 15 03 58 60 a9 00 8d  12
c9c8 : 02 dc 8d 27 d0 ad 00 dc  b5
c9d0 : 85 f9 a9 ff 8d 02 dc a5  64
c9d8 : c5 c9 03 d0 0b 20 3c c4  89
c9e0 : a9 00 8d 15 d0 4c 31 ea  99
c9e8 : c9 04 d0 0b 20 2a c4 a9  03
c9f0 : 01 8d 15 d0 4c 31 ea c9  a5
c9f8 : 05 d0 0b 20 33 c4 a9 00  2c
ca00 : 8d 15 d0 4c 31 ea c9 06  73
ca08 : d0 08 20 3c c4 a9 00 8d  21
ca10 : 15 d0 4c 31 ea 20 fd ae  cc
ca18 : 20 9e b7 20 50 f2 a9 00  bc
ca20 : a0 e0 85 fd 84 fe a2 19  4e
ca28 : a0 0d 2c a0 05 b9 98 ca  84
ca30 : 20 d2 ff 88 10 f7 a9 28  82
ca38 : 85 15 a9 80 85 97 a9 00  7e
ca40 : 85 14 a0 07 a9 34 78 85  01
ca48 : 01 b1 fd 85 02 a9 37 85  a7
ca50 : 01 58 a5 02 25 97 f0 07  08
ca58 : a5 14 19 a6 ca 85 14 88  5d
ca60 : 10 e2 a5 14 20 d2 ff 46  f2
ca68 : 97 90 d3 a5 fd 69 07 85  43
ca70 : fd 90 02 e6 fe c6 15 d0  2f
ca78 : c1 a9 0d 20 d2 ff a9 0a  3d
ca80 : 20 d2 ff ca d0 a5 a9 1b  7a
ca88 : 20 d2 ff a9 32 20 d2 ff  b6
ca90 : a9 0d 20 d2 ff 4c cc ff  b8
ca98 : 01 40 04 2a 1b 09 0d 14  56
caa0 : 00 65 1b 17 33 1b 80 40  8b
caa8 : 20 10 08 04 02 01 5e 5e  b1
```

Listing 3. Die Maschinenroutine zu »3D-Supergrafik«. Dieses Listing müssen Sie mit dem MSE eingeben.



Räumliche Grafik auf dem C 64

Gerade auf dem Gebiet von Film- und Fernsehproduktionen werden Computergrafiken immer öfter eingesetzt, da es hiermit möglich ist, Dinge zu zeigen, die mit herkömmlichen Methoden nicht gefilmt werden konnten.

Ob nun Raumschiffe gezeigt werden, die eigentlich unmögliche Bewegungen durchführen oder der »Anflug« eines Logos (wie zum Beispiel die »1« bei der ARD), die Realisierung vereinfacht sich durch den Einsatz von Grafik-Computern enorm. Diese müssen jedoch — bedingt durch die anfallenden Datenmengen — enorm schnell sein, so daß zur Zeit nur »Supercomputer« Echtzeitsimulation erlauben.

Wenn man sich allerdings etwas mehr Zeit läßt, dann reicht auch ein Homecomputer wie der C 64 aus, um solche Grafiken (mit bestimmten Einschränkungen) zu erzeugen.

Das hier vorgestellte Grafikpaket geht über das übliche Punkte setzen und Linien ziehen weit hinaus. Es erlaubt die Erzeugung von plastischen Bildern ohne größeren Aufwand, wie zum Beispiel in Bild 1 und 2 dargestellt.

Das eigentliche Grafikpaket nimmt nur 3 KByte (!) des Speichers in Anspruch und steht im Bereich ab $C000. Damit bleibt der Basic-Speicher für die eigentlichen Programme frei. Auch das DOS 5.1 von Commodore kann parallel zu den Grafikroutinen verwendet werden. Aufgrund der Kompaktheit dieses Grafikpakets mußten allerdings bestimmte Dinge, wie zum Beispiel die Abfrage auf legale Koordinaten, in den Hintergrund treten; die Software fängt also solche Fehler nicht ab. Das macht das Schreiben eines eigenen Programms zwar etwas komplizierter, hat man aber erst einmal alle Fehler beseitigt, so würde eine Fehlerabfrage den Ablauf nur verlangsamen.

Beginnen wir mit den Grundlagen zur Erzeugung der Grafik. Da stellt sich zunächt einmal die Frage, wie man die verschiedenen Helligkeitswerte und Schattierungen eines darzustellenden Objektes berechnet. Danach müssen diese verschiedenen Grauwerte in einzelne Punkte umgesetzt werden, denn auf dem HiRes-Bildschirm können diese ja nur gesetzt und gelöscht werden. Dazu muß die gesamte Software selbst geschrieben werden, da der C 64 diese Art der Grafik vom Betriebssystem her nicht unterstützt.

Schnelle Rechenroutinen

Das Maschinenprogramm, das für den Aufbau der Grafik zuständig ist, besteht aus mehreren Unterprogrammen. Beginnen wollen wir mit einem Unterprogrammpaket, das für schnelles Rechnen notwendig ist.

Integer Arithmetik (Listing 1)

Diese Unterroutinen stellen schnelle Algorithmen für Multiplikation, Division, Wurzelziehen und Zufallszahlen zur Verfügung. Diese sind zwar schon im normalen Basic-Interpreter vorhanden, für unsere Anwendung sind sie jedoch aus Geschwindigkeitsgründen ungeeignet. Die Basic-Routinen sind nämlich nicht auf Geschwindigkeit, sondern auf Kürze optimiert worden, wie zum Beispiel die Funktion SQR. Die

Wurzelberechnung in Basic teilt den Logarithmus durch 2 und potenziert das Ergebnis anschließend wieder. Das hat den Vorteil, daß der Interpreter auf bereits vorhandene Unterprogramme zugreifen kann, aber den Nachteil, daß diese Berechnungsweise relativ langsam ist. Wenn man für alle 64000 Bildschirmpunkte die Quadratwurzel mit den vorhandenen Routinen berechnen würde, ergäbe sich eine Zeichenzeit von mehr als einer Stunde (und dies trotz Maschinenspracheprogramm)! Man benötigt diese Berechnung zwar nicht für jeden Punkt, aber gerade bei der Berechnung der Schattierungswerte spielt die Quadratwurzel eine große Rolle. Eine Möglichkeit, die Wurzelberechnung zu beschleunigen bietet die Newton-Methode, mit der man Fließkommawurzeln in

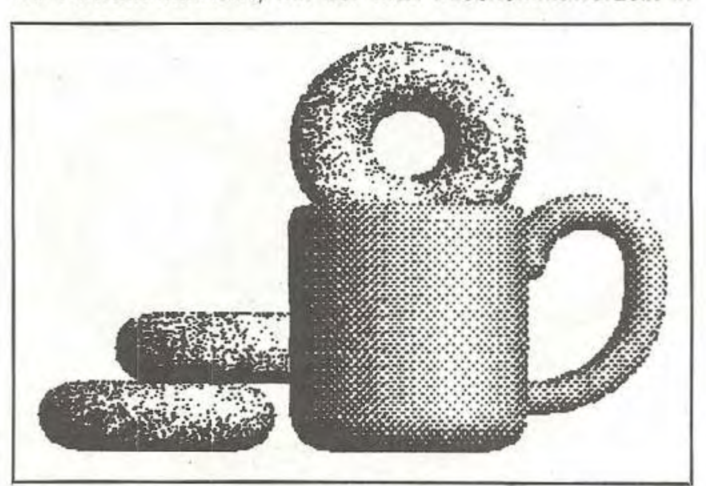


Bild 1. Eindrucksvolle räumliche Wirkung durch Kombination verschiedener Shapes

einem Bruchteil der sonst benötigten Zeit berechnen kann. Da wir, um die einzelnen Grafikpunkte zu adressieren, nur ganze Zahlen benötigen, beschleunigt sich die Abarbeitungszeit nochmals. Auch die Routine für die Erzeugung von Zufallszahlen, die später bei der »Random«-Schattierung erforderlich ist, wurde neu geschrieben.

Grafik Utilities (Listing 2)

Die nächste Unterprogrammsammlung beinhaltet Basisroutinen für die Verwaltung der Bitmap wie zum Beispiel löschen, Farbe setzen und Punkte plotten. Diese Unterroutinen sind als einzige Commodore 64-spezifisch, so daß man — möchte man das Programm auf andere 6502-Computer übertragen — nur diese an das jeweilige Gerät anpassen muß.

Ferner wurden noch zwei Schattierungsroutinen »SHADE« und »RSHADE« implementiert. »SHADE« arbeitet im Prinzip ganz einfach: Man definiert eine 8 x 8 Graustufenmatrix, in der Werte von 0 bis 63 enthalten sind (Tabelle 1). Mit Hilfe dieser Daten wird nun ein — ebenfalls 8 x 8 Pixel großes — Feld erstellt, daß dann an einer bestimmten Stelle in den HiRes-Bildschirm kopiert wird. Um eine gleichmäßige Grauschattierung zu erreichen, müssen die Pixels auch möglichst einen gleichmäßigen Abstand voneinander haben. Wenn das Programm nun einen Grauwert in die Grafik setzen will, so übergibt es einfach einen Wert zwischen 0 und 63 an diese Unterroutine. Nun werden innerhalb dieser Matrix die einzelnen Punkte der Reihe nach gesetzt, also zuerst der Punkt mit der Nummer 0 (oben links), dann 1 (4. von links und 4. von oben) und so weiter. Also, je größer die Zahl ist, desto mehr Punkte werden gesetzt (und um so dunkler wird das Ganze). Durch diese festgelegte Reihenfolge wird sichergestellt, daß die Schattierung gleichmäßig ist. Die RSHADE-Unterroutine hat genau die gleiche Aufgabe, sorgt aber für etwas ungleichmäßigere — durch den Zufallsgenerator gesteuerte — Grauwerte.

Ein weiteres zu lösendes Problem ist das etwas merkwürdige rechteckige Bildschirmformat des C 64. Die Auflösung von 320 x 200 Punkten entspricht einem Teilungsverhältnis von 8:5, wodurch es bei einem normalen Bildschirm — mit einem Verhältnis von 4:3 — zu einer horizontalen Verzerrung kommen kann (eine Kugel würde dann etwa wie ein plattgedrückter Ball aussehen). Die Unterroutine »SCALE« (in Listing 2) sorgt für diese darstellungs- und maßstabsgerechte Aufbereitung der Koordinaten.

Linien und Facettenzeichnen

Die Routinen in Listing 3 komplettieren die elementaren Grafik-Funktionen für das Ziehen von Linien und das Zeichnen von schattierten Polygonnetzen. Ein solches räumliches Polygonnetz ist in Bild 3 zu sehen; es besteht nur aus aneinander gesetzten und entsprechend schattierten Dreiecken. Diese Figuren können einfacher und schneller gezeichnet werden, da die Schattierung auf ebenen Flächen einfacher darzustellen ist als auf gekrümmten. Neben der üblichen Darstellungsweise (in der geschlossenen, schattierten Form) ist auch die Zeichnung von Grafiken ohne verdeckte Linien möglich.

Mit diesen Routinen gerüstet, können wir uns nun auf das eigentliche Shape-Unterprogramm konzentrieren, mit dem

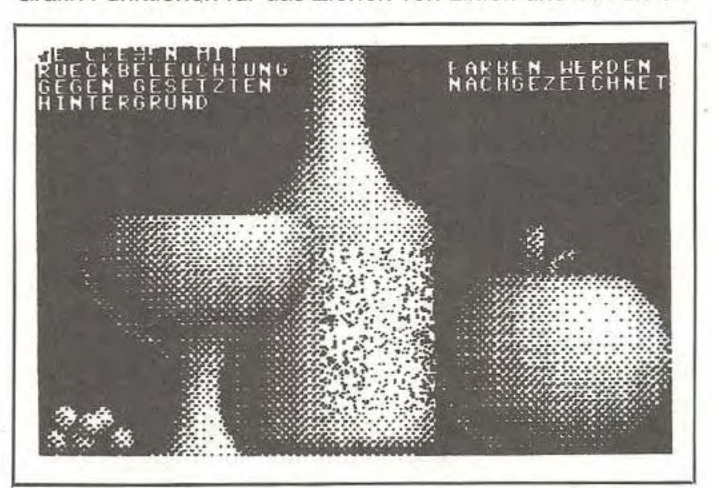


Bild 2. Verblüffende Effekte lassen sich erzielen. Beachten Sie den Lichteinfall.

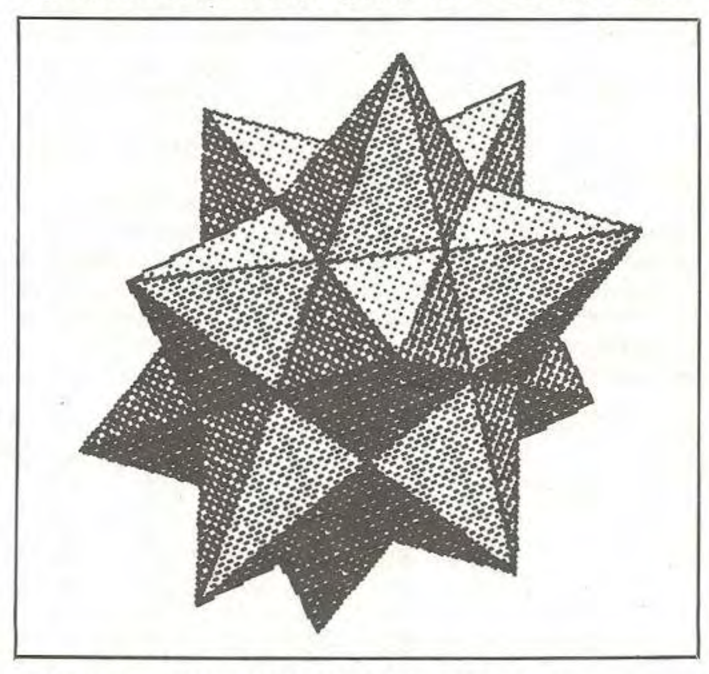


Bild 3. Ein weiteres Beispiel für die 3D-Wirkung

es möglich ist, gekrümmte, schattierte Figuren in kurzer Zeit zu erstellen.

Hier geht es wiederum als erstes um die Berechnung der verschiedenen Oberflächenreflexe, Brechungen, Streuungen und Absorptionen des Lichtes auf einer Oberfläche. Um diese, teilweise sehr komplizierten Berechnungenn, zu vereinfachen, geht man einen anderen Weg: Man beschränkt sich auf einige einfache symmetrische Figuren (Bild 4), aus denen alle zu zeichnenden Objekte zusammengesetzt werden. Weiterhin werden die Figuren nur in der Frontsicht, also von vorn, dargestellt; Drehungen sind nicht möglich. Durch diese Einschränkungen braucht man bei der Programmierung nicht auf komplizierte Vektorberechnungen zurückzugreifen.

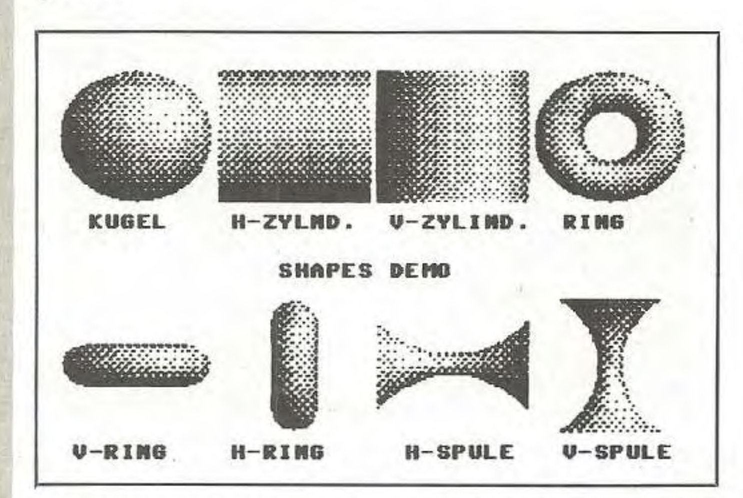


Bild 4. Die zur Verfügung stehenden Shapes, aus denen alle Figuren zusammengesetzt werden

Für die Berechnung des Lichteinfalls stelle man sich einfach eine Lampe vor, die an der rechten Schulter des Beobachters angebracht ist und das Objekt von dieser Position anleuchtet. Dementsprechend errechnet das Programm die Helligkeitswerte für jeden Punkt des Objektes. Ein Flag (Adresse 898) dient dazu, den Standort der »Lampe« (0= Frontbeleuchtung, 1= Beleuchtung von hinten) festzulegen.

Zeichenroutinen für Shapes

Listing 4 beinhaltet eine Sammlung von Routinen, die die acht Grundshapes (siehe Bild 4) zeichnen. Da dieses Unterprogramm auf die — an den C 64 angepaßten — Lower-Level-Routinen der Grafik-Utilities (Listing 2) zugreift, ist es selbst nicht vom jeweiligen Computer abhängig. Zusammen mit dem, in Listing 5 abgebildeten Programm, wird der Kontakt zum Basic hergestellt. Die Übergabe der Parameter geschieht mittels »SYS«-Befehl:
SYS <KOMMANDO>, <PARAM. 1,>, <PARAM. 2>, ..., wobei <KOMMANDO> die Adresse des jeweiligen Grafikbefehls ist (Tabelle 2 zeigt die komplette Befehlsübersicht dieses Grafikprogramms). Über <PARAMETER 1> und <PARAMETER 2> müssen die Koordinaten des Shapes angegeben werden, wobei zu beachten ist, daß der Ursprung (dort wo die X- und Y-Koordinate null ist) in der linken unteren Bildschirmecke liegt. Die einzusetzenden Werte gehen in der Horizontalen von 0 bis 255, in der Vertikalen von 0 bis 239.

Alle Shapes, mit Ausnahme der Kugel, können durch die Eingabe von insgesamt vier Parametern in ihrer Position und Form beeinflußt werden. Die ersten zwei bestimmen, wie oben erläutert, die Koordinaten des Figurmittelpunktes. Mit dem dritten und vierten Paramter kann man die Verzerrung in X- und Y-Richtung einstellen. Doch hier ist Vorsicht geboten, denn der X-Wert sollte hierbei immer größer als der Y-Wert sein, sonst zeichnet das Programm unerwünschte Figuren.

Eine weitere Besonderheit sind die Shape-Fenster. Damit ist es möglich, auch einzelne Teile eines Shapes auf dem Bildschirm zu bringen. Diese Funktion wurde zum Beispiel bei der Erstellung der Grafik in Bild 1 verwendet. Der Henkel die-

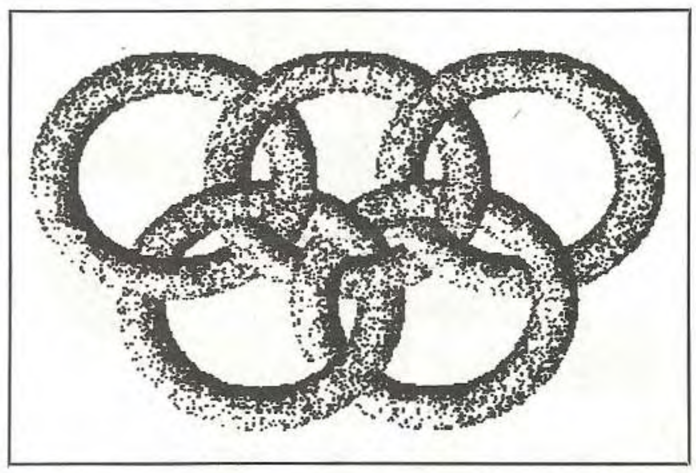


Bild 5. Aus einfachen Figuren lassen sich verblüffend »lebendige« Bilder zusammensetzen

0	8	53	61	2	10	55	63
16	24	37	45	18	26	39	47
49	57	4	12	51	59	6	14
33	41	20	28	35	43	22	30
3	11	54	62	1	9	52	60
19	27	38	46	17	25	36	44
50	58	7	15	48	56	5	13
34	42	23	31	32	40	21	29

Tabelle 1. Matrix der Grauwerte

Grafikspeicher: 40960-48959
Farbspeicher: 33792-34791

Shape-Fenster
893: Linker Rand
894: Rechter Rand
895: Unterer Rand
896: Oberer Rand

Style Flags
838: Schattierungsart (0 = zufällig, 1 = Halbton)
839: Skalierungsflag (0 = normal 1:1, 1 = skaliert 4:3)
868: Eck Flag bei Polygonen (0 = Normal, 1 = zeichnet Linien an den Ecken)
871: Eckenmodus (0 = zeichnet Linie, 1 = löscht Rand)
898: Beleuchtungsflag (0 = von vorne, 1 = von hinten)

Funktionsadressen
49378: Auf den Grafikmodus umschalten
49411: Zurück zum Textbildschirm
51979: Bitmap löschen
52001: Bitmap färben
52023: PLOT (Parameter X,Y) setzt einen Punkt
52026: UNPLOT löscht einen Punkt mit den Koordinaten X und Y
52049: LINE (Parameter X1,Y1,X2,X2) zieht Linie von / bis
52052: FACET X1,Y1,X2,Y2,X3,Y3,SA zeichnet dreieckige Facetten mit den Koordinaten X1,Y1, X2,... und schattiert sie (SA: 0 = schwarz, 64 = weiß)

Shapes
52119: Kugel
52141: Ring
52150: vertikaler Zylinder
52153: horizontaler Zylinder
52186: vertikaler Ring
52189: horizontaler Ring
52203: vertikale Rolle
52206: horizontale Rolle

Tabelle 2. Befehlsübersicht

ser Kaffeetasse ist aus drei verschiedenen Shapes zusammengesetzt, die mit Hilfe des Fensters in ihrer Ausdehnung eingeschränkt wurden. Diesen Begrenzungsfenstern sind die Adressen 893 bis 896 zugeordnet, wobei sich die POKE-Werte auf den Mittelpunkt der Figur beziehen.

Die Kommandos, um die Bitmap zu löschen oder mit Punkten zu füllen, benötigen auch alle nur einen Parameter. So sieht zum Beispiel das Kommando zum Initialisieren des Farbspeichers so aus:

POKE 52001,16*DC+BC

wobei DC die Farbe der zu setzenden Punkte und BC die Hintergrundfarbe ist. Auch Texte können nach Wunsch auf dem Grafikbildschirm ausgegeben werden. Die dafür zuständigen Routinen sind in Listing 2 enthalten.

Die in Listing 10 und 11 abgedruckten Demonstrationsprogramme sollen helfen, die Arbeitsweise und die Kombinationsmöglichkeiten der einzelnen Shapes aufzuzeigen. Auch alle hier gezeigten Bilder stammen von diesem Programm. Die Bilder stehen im Speicher übrigens »unter« dem Basic-ROM im Bereich zwischen $A000 und $BFFF.

Um den Platz für das DOS 5.1 freizuhalten, wurde das Grafikpaket in zwei Teile aufgespalten. Der eine Teil mit dem Namen GRAPHICS (Listing 8) befindet sich im Bereich zwischen $C000 und $CBFC. Der zweite Teil, auf den im folgenden noch kurz eingegangen werden soll, muß nach Adresse $CF59 geladen werden. Dieser Programmteil enthält eine schnelle Sortierroutine (Listing 6) und ermöglicht die Darstellung von Text auf dem Grafikbildschirm (Listing 7). Die beiden Unterprogramme sind in dem MSE-Listing 9 zusammengefaßt worden und müssen gemeinsam mit »GRAPHICS« in den Speicher geladen werden.

Hinweise zum Abtippen

Sie müssen als erstes die MSE-Listings 8 und 9 eingeben und anschließend natürlich auch abspeichern. Die Listings 1 bis 7 dienen nur zur Dokumentation und brauchen nicht eingegeben werden. Nachdem Sie auch die Demos abgetippt und auf Diskette oder Band gespeichert haben, müssen die einzelnen Programmteile nacheinander in den Speicher geladen werden. Begonnen wird mit GRAPHICS, das mit LOAD"GRAPHICS",8,1 in den Speicher gelesen werden muß. Nachdem man anschließend NEW eingegeben hat, muß das zweite Maschinenprogramm, wie oben, geladen werden (auch hier das NEW nicht vergessen). Anschließend kann wahlweise eines der beiden Demonstrationsprogramme (Listing 10 oder 11) geladen und ganz normal gestartet werden.

(Richard Rylander/Christoph Sauer/ev)

```
; INTEGER ARITHMETIC ROUTINES
;
; RICHARD L. RYLANDER    8/12/84
;
; REVISED 10/29/84 TO ADD FULL DOUBLE
; PRECISION ARGUMENTS IN DIVIDE ROUTINE
;
;*****************************************
;
; USE PAGE ZERO LOCATIONS WHERE POSSIBLE FOR
; ITERATIVE PROCEDURE WORK SPACE
;
MLPCND =$AC     ; MULTIPLICAND
MLPLER =$AD     ; MULTIPLIER
PROD   =$AE     ; PRODUCT
;
DVDND  =$FD     ; DIVIDEND/QUOTIENT
DVSOR  =$FB     ; DIVISOR
RMNDR  =$B4     ; REMAINDER
;
RADCND =$AC     ; RADICAND
ROOT   =$033C   ; SQUARE ROOT
;
TEMP   =$FB
;
; SET UP SEED VALUES FOR PSEUDO-RANDOM NUMBERS
       *=$C000
RNDM   .BYTE $FF,$55
RTEMP  .BYTE $00,$00
;
;
;*****************************************
;
; MULTIPLY SINGLE PRECISION MULTIPLICAND
; BY SINGLE PRECISION MULTIPLIER GIVING
; DOUBLE PRECISION PRODUCT (ENTER AT "MULT")
;
; SPECIAL CASE: ENTER AT "SQUARE" TO FIND
; SQUARE OF SIGNED 8-BIT NUMBER
;
SQUARE LDA MLPCND  ; ENTRY TO SQUARE
       BPL POSITV  ; USE ABSOLUTE VALUE
       SEC         ; NEGATE IF NEEDED
       LDA #$00
       SBC MLPCND
       STA MLPCND
POSITV STA MLPLER
MULT   LDA #$00    ; ENTRY TO MULTIPLY
       LDX #$08
MLOOP  LSR MLPLER
       BCC NOADD
       CLC
       ADC MLPCND
NOADD  ROR A
       ROR PROD
       DEX
       BNE MLOOP
       STA PROD+1
       RTS
;
;*****************************************
;
; DIVIDE DOUBLE PRECISION DIVIDEND
; BY DOUBLE PRECISION DIVISOR GIVING
; DOUBLE PRECISION QUOTIENT
;
; DIVIDEND IS REPLACED BY QUOTIENT
; IN THE PROCESS
;
; QUOTIENT IS ROUNDED TO NEAREST INTEGER
;
DIVIDE LDA #$00
       STA RMNDR
       STA RMNDR+1
       LDX #$10
DLOOP  ROL DVDND
       ROL DVDND+1
       ROL RMNDR
       ROL RMNDR+1
       SEC
       LDA RMNDR
       SBC DVSOR
       TAY
       LDA RMNDR+1
       SBC DVSOR+1
       BCC DECCNT
       STY RMNDR
       STA RMNDR+1
DECCNT DEX
       BNE DLOOP
       ROL DVDND     ; CHECK IF REMAINDER
       ROL DVDND+1   ; IS >= 1/2 OF DIVIDEND
       ASL RMNDR     ; FOR ROUNDING
       ROL RMNDR+1
       BCS ROUND
       SEC
       LDA DVSOR
       SBC RMNDR
       LDA DVSOR+1
       SBC RMNDR+1
       BCS NOCHNG
ROUND  INC DVDND
       BNE NOCHNG
       INC DVDND+1
NOCHNG RTS
;
;*****************************************
;
; TAKE INTEGER SQUARE ROOT OF A
; DOUBLE PRECISION RADICAND GIVING
; SINGLE PRECISION ROOT ( <= REAL ROOT )
;
SQRT   LDX #$08
       LDA #$00
       STA ROOT
       STA ROOT+1
       STA TEMP
       STA TEMP+1
SQRT1  ASL ROOT
       ROL ROOT+1
       INC ROOT      ; ASSUME CURRENT LSB OF
       BNE NEXT1     ; ROOT WILL BE 1
       INC ROOT+1
NEXT1  ASL RADCND    ; SHIFT RADICAND LEFT
       ROL RADCND+1  ; TWICE INTO TEMP
       ROL TEMP
       ROL TEMP+1
       ASL RADCND
       ROL RADCND+1
       ROL TEMP
       ROL TEMP+1
       SEC           ; SUBTRACT ROOT ESTIMATE
       LDA TEMP      ; FROM TEMP
       SBC ROOT
       TAY
       LDA TEMP+1
       SBC ROOT+1
       BCC RESTOR
       STA TEMP+1    ; SUBTRACTION OK
       STY TEMP
       INC ROOT
       BNE NEXT2
       INC ROOT+1
NEXT2  DEX
       BNE SQRT1
       JMP FINI
RESTOR SEC           ; IGNORE SUBTRACTION
       LDA ROOT      ; AND RESET LSB OF ROOT
       SBC #$01
       STA ROOT
       BCS NEXT3
       DEC ROOT+1
NEXT3  DEX
       BNE SQRT1
FINI   ROR ROOT+1    ; FINAL /2 TO NORMALIZE
       ROR ROOT
       RTS
;
;*****************************************
;
; GENERATE PSEUDO-RANDOM BYTES
; EXIT WITH P-R BYTE IN ACCUM.
;
RANDOM LDA RNDM
       STA RTEMP
       EOR RNDM+1
       ROL RTEMP+1   ; RTEMP+1 PRESERVES
       ROR A         ; CARRY BIT FOR CYCLING
       ROR RTEMP+1   ; RANDOM NUMBERS
       STA RNDM
       LDA RTEMP
       STA RNDM+1
       RTS
.END
```

Listing 1. Assembler-Quelltext der schnellen Integer-Arithmetik

```
; GRAPHICS UTILITIES
;
; RICHARD L. RYLANDER    11/4/84
;
; LOAD ARITHMETIC UTILITIES FIRST
;
RAM=$033E
ORIGIN=$C0E2
;
MLPCND=$AC      ; MULTIPLICAND (S)
MLPLER=$AD      ; MULTIPLIER (S)
PROD=$AE        ; PRODUCT (D)
MULT=$C011      ; CALL FOR MULTIPLY
;
RNDM=$C000      ; RANDOM NUMBER
RANDOM=$C0C8    ; CALL FOR RANDOM
; NOTE - A CALL TO 'RANDOM' LEAVES A RANDOM BYTE
; IN THE ACCUMULATOR
;
       *=RAM
PLTFLG *=*+1    ; PLOT/UNPLOT FLAG
XPLT   *=*+2    ; ABSOLUTE PLOT COORD
YPLT   *=*+1    ; ABSOLUTE PLOT COORD
VIC1   *=*+1    ; REGISTER STORAGE
VIC2   *=*+1    ; REGISTER STORAGE
VALUE  *=*+2    ; FINAL NORMALIZED SHADE VALUE
HTORRN *=*+1    ; SHADE FLAG, 1=HALFTONE
NOSCAL *=*+1    ; SCALE FLAG, 1=NO SCALE
TEMP   *=*+2    ; TEMPORARY STORAGE
;
       *=ORIGIN
;
;*****************************************
;
; TURN ON BIT MAP GRAPHICS MODE,
; SAVING REGISTER VALUES FOR
; RETURN TO TEXT MODE LATER.
;
GRFON  LDA $D011
       ORA #$20
       STA $D011
       LDA $DD00
       STA VIC1
       AND #$FC
       ORA #$01
       STA $DD00
       LDA $D018
       STA VIC2
       LDA #$19
       STA $D018
       RTS
;
;*****************************************
;
; RETURN TO TEXT SCREEN
;
GRFOFF LDA $D011
       AND #$DF
       STA $D011
       LDA VIC1
       STA $DD00
       LDA VIC2
       STA $D018
       RTS
;
;*****************************************
;
; FILL COLOR MAP FOR BLACK DOTS ON WHITE
;
COLOR  LDA #$01     ; POKE NEW COLORS HERE
       LDX #0
COL1   STA $8400,X
       STA $8500,X
       STA $8600,X
       STA $8700,X
       DEX
       BNE COL1
       RTS
;
;*****************************************
;
; CLEAR HI-RES GRAPHICS SCREEN
;
CLEAR  LDA #$A0
       STA $FC
       LDY #0
       STY $FB
       LDA #0       ; CLEAR BYTE
```

Listing 2. Assembler-Quelltext der Grafik-Utilities

```
        LDX #$20
CLRLP   STA ($FB),Y
        INY
        BNE CLRLP
        INC $FC
        DEX
        BNE CLRLP
        RTS
;
;*****************************************
;
; PLOT AND UNPLOT POINTS ON HI-RES GRAPHICS
; SCREEN. ABSOLUTE X AND Y SCREEN COORDINATES
; ARE POKED INTO XPLT, XPLT+1, AND YPLT
;
PLOT    LDA #0
        .BYTE $2C
UNPLOT  LDA #$80
        STA PLTFLG
        LDA $01       ; BASIC ROM OUT
        AND #$FE
        STA $01
        SEC           ; INVERT Y COORDINATE TO
        LDA #$C7      ; PUT ORIGIN IN LOWER LEFT
        SBC YPLT      ; CORNER OF SCREEN
        TAX           ; (199.-YPLT)
        LSR A
        LSR A
        LSR A
        TAY
        LDA TABLE1,Y
        STA $FB
        LDA TABLE2,Y
        STA $FC
        TXA
        AND #$07
        CLC
        ADC $FB
        STA $FB
        LDA XPLT
        AND #$F8
        ADC $FB
        STA $FB
        LDA XPLT+1
        ADC $FC
        STA $FC
        LDA #$A0
        ADC $FC
        STA $FC
        LDA XPLT
        AND #$07
        EOR #$07
        TAX
        LDA #$01
PLOTLP  DEX
        BMI PLOT2
        ASL A
        BNE PLOTLP
PLOT2   LDY #0
        BIT PLTFLG
        BPL NOPLOT
        EOR #$FF
        AND ($FB),Y
        .BYTE $2C
NOPLOT  ORA ($FB),Y
        STA ($FB),Y
        LDA $01       ; BASIC ROM RESTORED
        ORA #$01
        STA $01
        RTS
;
TABLE1 .BYTE $00,$40,$80,$C0
       .BYTE $00,$40,$80,$C0
       .BYTE $00,$40,$80,$C0
       .BYTE $00,$40,$80,$C0
       .BYTE $00,$40,$80,$C0
       .BYTE $00,$40,$80,$C0,$00
;
TABLE2 .BYTE $00,$01,$02,$03
       .BYTE $05,$06,$07,$08
       .BYTE $0A,$0B,$0C,$0D
       .BYTE $0F,$10,$11,$12
       .BYTE $14,$15,$16,$17
       .BYTE $19,$1A,$1B,$1C,$1E
;
;*****************************************
;
; SHADING BY HYBRID DITHER/DOT-GROWTH
;
SHADE   LDA XPLT      ; USE BITS -----***
        AND #$07      ; OF 'X' SCREEN COORD
        STA TEMP
        LDA YPLT      ; AND BITS -----***
        AND #$07      ; OF 'Y' SCREEN COORD
        ASL A         ; SHIFTED INTO --***---
        ASL A         ; POSITION TO DETERMINE
        ASL A         ; 6-BIT OFFSET IN
        ORA TEMP      ; THRESHOLD TABLE
        TAX
        LDA THRESH,X  ; SCREEN-POSITION-WEIGHTE
        CMP VALUE     ; THRESHOLD VALUE
        BPL GREATR
        JMP UNPLOT
GREATR  JMP PLOT
;
;*****************************************
;
; SHADING BY RANDOM HALFTONE
;
RSHADE  JSR RANDOM
        LSR A          ; REDUCE RANDOM BYTE
        LSR A          ; TO 6 BITS FOR SHADE
        CMP VALUE      ; VALUE COMPARISON
        BPL MORE
        JMP UNPLOT
MORE    JMP PLOT
;
;*****************************************
;
; PLOT A POINT WEIGHTED BY SHADING SCHEME
; AND SHADE VALUE
; CHECK 'NOSCAL' FLAG FOR SCALING OF Y COORD
; CHECK 'HTORRN' FLAG FOR TYPE OF SHADING
;
PLTSHD  LDA NOSCAL
        BEQ NORM
;
; SCALE Y FROM 0-239 PSEUDO-COORDINATES
; TO 0-199 TRUE SCREEN COORDINATES BY
;     Y = (Y+1)*213/256
;
SCALE   LDY YPLT
        INY
        STY MLPLER
        LDA #$D5       ; 213.
        STA MLPCND
        JSR MULT       ; RETURN WITH HIGH BYTE
        STA YPLT       ; IN ACCUMULATOR
NORM    LDA HTORRN
        BEQ RPLT
        JMP SHADE
RPLT    JMP RSHADE
;
;
;
```

```
THRESH .BYTE $00,$08,$35,$3D
       .BYTE $02,$0A,$37,$3F
       .BYTE $10,$18,$25,$2D
       .BYTE $12,$1A,$27,$2F
       .BYTE $31,$39,$04,$0C
       .BYTE $33,$3B,$06,$0E
       .BYTE $21,$29,$14,$1C
       .BYTE $23,$2B,$16,$1E
       .BYTE $03,$0B,$36,$3E
       .BYTE $01,$09,$34,$3C
       .BYTE $13,$1B,$26,$2E
       .BYTE $11,$19,$24,$2C
       .BYTE $32,$3A,$07,$0F
       .BYTE $30,$38,$05,$0D
       .BYTE $22,$2A,$17,$1F
       .BYTE $20,$28,$15,$1D
.END
```

Listing 2. Assembler-Quelltext der Grafik-Utilities (Schluß)

```
; FACET - DRAW SHADED TRIANGULAR FACETS
; AND STRAIGHT LINES.
;
; RICHARD L. RYLANDER   11/4/84
;
; LOAD "ARITH.HEX" AND "GRAPH.HEX"
; BEFORE USING
;
ORIGIN = $C26F
RAM    = $034A
;
XPLT   = $033F
YPLT   = $0341
NORM   = $C224
NOSCAL = $0347
PLOT   = $C143
UNPLOT = $C146
;
MLPCND = $AC
MLPLER = $AD
PROD   = $AE
MULT   = $C011
;
DVDND  = $FD
DVSOR  = $FB
QUOT   = $FD
DIVIDE = $C025
;
      *=RAM
;
XMIN   *=*+2
YMIN   *=*+1
XMID   *=*+2
YMID   *=*+1
XMAX   *=*+2
YMAX   *=*+1
YTOP   *=*+1
YBOT   *=*+1
YBASE  *=*+1
DLTAX1 *=*+2
DLTAX2 *=*+1
DLTAX3 *=*+1
DELTAX *=*+1
DLTAY1 *=*+1
DLTAY2 *=*+1
DLTAY3 *=*+1
DELTAY *=*+1
XDIFF  *=*+1
FLAG1  *=*+1
FLAG2  *=*+1
FLAG3  *=*+1
FLAG   *=*+1
EDGES  *=*+1
ERROR  *=*+2
MODE   *=*+1
COUNT  *-*+2
;
;
      *=ORIGIN
;
;*****************************************
;
; SCALE ALL Y COORDINATES FROM 0..239
; PSUEDO-COORDINATE RANGE TO 0..199
; TRUE SCREEN COORDINATE RANGE
;
SCALE   LDY #6
        LDA #$D5
        STA MLPCND
SCLP    LDA YMIN,Y
        STA MLPLER
        JSR MULT
        STA YMIN,Y
        DEY
        DEY
        DEY
        BPL SCLP
        RTS
;
;
;*****************************************
;
; EXCHANGE 'MIN' AND 'MID' COORDINATES
;
SWAP12  LDY #2
LOOP1   LDA XMIN,Y
        PHA
        LDA XMID,Y
        STA XMIN,Y
        PLA
        STA XMID,Y
        DEY
        BPL LOOP1
        RTS
;
;*****************************************
;
; EXCHANGE 'MID' AND 'MAX' COORDINATES
;
SWAP23  LDY #2
LOOP2   LDA XMID,Y
        PHA
        LDA XMAX,Y
        STA XMID,Y
        PLA
        STA XMAX,Y
        DEY
        BPL LOOP2
        RTS
;
;*****************************************
;
; SORT COORDINATES ACCORDING TO X COMPONENTS
;
SORTX   LDX #2
SORTLP  SEC
        LDA XMID
        SBC XMIN
        LDA XMID+1
        SBC XMIN+1
        BCS NOSWP1
        JSR SWAP12
NOSWP1  DEX
        BEQ SORTED
        SEC
        LDA XMAX
        SBC XMID
        LDA XMAX+1
        SBC XMID+1
        BCS SORTLP
        JSR SWAP23
        JMP SORTLP
SORTED  RTS
;
;*****************************************
;
; DRAW A LINE BETWEEN XMIN,YMIN AND XMID,YMID
; USING FAST DDA (DIGITAL DIFFERENTIAL ANALYZER)
; TECHNIQUE
;
LINE    LDA #2         ; ENSURE XMAX IS
        STA XMAX+1     ; LARGEST BEFORE
        JSR SORTX      ; ORDERING 'MIN' AND 'MID'
        LDA NOSCAL
        BEQ OUTLN
        JSR SCALE
OUTLN   JSR FINDXY     ; ENTRY POINT TO
        LDA XMIN        ; OUTLINE FACETS
        STA XPLT
        LDA XMIN+1
        STA XPLT+1
        LDA YMIN
        STA YPLT
        LDA DLTAX1+1   ; CHECK FOR DX>DY
        BNE STEPX
        SEC
        LDA DLTAX1
        SBC DLTAY1
        BCS STEPX
STEPY   LDA DLTAY1
        STA ERROR
        STA COUNT
        LSR ERROR
        SEC
        LDA DLTAX1
        SBC ERROR
        STA ERROR
        LDA DLTAX1+1
        SBC #0
        STA ERROR+1
        INC COUNT
LNLP1   LDA MODE       ; 0 = DRAW, 1 = ERASE
        BNE ERASE1
        JSR PLOT
        JMP SK1
ERASE1  JSR UNPLOT
SK1     LDA FLAG1      ; 0 = POSITIVE SLOPE
        BNE NSLOPE
        INC YPLT
        BNE SK2       ; ALWAYS BRANCH
NSLOPE  DEC YPLT
SK2     BIT ERROR+1
        BMI SK3
        INC XPLT
        BNE NOINC1
        INC XPLT+1
NOINC1  SEC
        LDA ERROR
        SBC DLTAY1
        STA ERROR
        LDA ERROR+1
        SBC #0
        STA ERROR+1
SK3     CLC
        LDA ERROR
        ADC DLTAX1
        STA ERROR
        LDA ERROR+1
        ADC DLTAX1+1
        STA ERROR+1
        DEC COUNT
        BNE LNLP1
        RTS
;
STEPX   LDA DLTAX1
        STA ERROR
        STA COUNT
        LDA DLTAX1+1
        STA ERROR+1
        STA COUNT+1
        LSR ERROR+1
        ROR ERROR
        SEC
        LDA DLTAY1
        SBC ERROR
        STA ERROR
        LDA #0
        SBC ERROR+1
        STA ERROR+1
LNLP2   LDA MODE
        BNE ERASE2
        JSR PLOT
        JMP SKP1
ERASE2  JSR UNPLOT
SKP1    INC XPLT
        BNE NOINC2
        INC XPLT+1
NOINC2  BIT ERROR+1
        BMI SKP3
        LDA FLAG1
        BNE NGSLP
        INC YPLT
        BNE SKP2      ; ALWAYS BRANCH
NGSLP   DEC YPLT
SKP2    SEC
        LDA ERROR
        SBC DLTAX1
        STA ERROR
        LDA ERROR+1
        SBC DLTAX1+1
        STA ERROR+1
SKP3    CLC
        LDA ERROR
        ADC DLTAY1
        STA ERROR
        LDA ERROR+1
        ADC #0
        STA ERROR+1
        SEC
        LDA COUNT
        SBC #1
        STA COUNT
        BCS TEST
        DEC COUNT+1
TEST    BIT COUNT+1
        BPL LNLP2
        RTS
```

Listing 3. Assembler-Quelltext zum Zeichnen von Linien und Dreiecken

```
;
;****************************************
; DRAW A SHADED VERTICAL LINE AT
; XPLT FROM YTOP TO YBOT
;
VLINE  SEC             ; MAKE SURE YTOP>YBOT
       LDA YTOP
       SBC YBOT
       BCS DRAW
       LDA YTOP
       PHA
       LDA YBOT
       STA YTOP
       PLA
       STA YBOT
DRAW   LDA YTOP
       STA YPLT
       JSR NORM        ; PLOT A SHADE-WEIGHTED
       LDA YTOP        ; PIXEL CHECKING ONLY
       CMP YBOT        ; FOR SHADE STYLE
       BEQ DONE
       DEC YTOP
       JMP DRAW
DONE   RTS
;
;****************************************
;
; FIND ENDPOINTS FOR VERTICAL LINES
; BETWEEN FACET EDGES
;
ENDPTS LDA XDIFF
       STA MLPCND
       LDA DELTAY
       STA MLPLER
       JSR MULT
       STA DVDND+1
       LDA PROD
       STA DVDND
       LDA #0
       STA DVSOR+1
       LDA DELTAX
       STA DVSOR
       JSR DIVIDE
       LDA FLAG
       BNE NEGSLP
       CLC
       LDA YBASE
       ADC QUOT
       BCC SKIP2
NEGSLP SEC
       LDA YBASE
       SBC QUOT
SKIP2  RTS
;
;****************************************
;
; FIND COORDINATE DIFFERENCES
;
; ALL "DELTA X" VALUES POSITIVE,
; SINGLE PRECISION (JUST LOWER BYTE)
;
FINDXY SEC
       LDA XMID
       SBC XMIN
       STA DLTAX1
       LDA XMID+1
       SBC XMIN+1
       STA DLTAX1+1
       SEC
       LDA XMAX
       SBC XMID
       STA DLTAX2
       SEC
       LDA XMAX
       SBC XMIN
       STA DLTAX3
;
; USE ABS(DELTA Y) VALUES,
; FLAGS INDICATE SLOPE OF LIMIT LINES
;
       LDA #$00
       STA FLAG1
       STA FLAG2
       STA FLAG3
       SEC
       LDA YMID
       SBC YMIN
       BCS STORE1
       INC FLAG1
       LDA YMIN
       SBC YMID
STORE1 STA DLTAY1
       SEC
       LDA YMAX
       SBC YMID
       BCS STORE2
       INC FLAG2
       LDA YMID
       SBC YMAX
STORE2 STA DLTAY2
       SEC
       LDA YMAX
       SBC YMIN
       BCS STORE3
       INC FLAG3
       LDA YMIN
       SBC YMAX
STORE3 STA DLTAY3
       RTS
;
;****************************************
;
; DRAW A SHADED TRIANGULAR FACET
;
FACET  JSR SORTX
       LDA NOSCAL
       BEQ YSOK
       JSR SCALE
YSOK   JSR FINDXY
       LDA XMIN
       STA XPLT
       LDA XMIN+1
       STA XPLT+1
FCETLP SEC
       LDA XPLT
       SBC XMIN
       STA XDIFF
       LDA DLTAX1
       BEQ CONT
       STA DELTAX
       LDA DLTAY1
       STA DELTAY
       LDA FLAG1
       STA FLAG
       LDA YMIN
       STA YBASE
       JSR ENDPTS
       STA YTOP
       LDA DLTAX3
       BEQ CONT
       STA DELTAX
       LDA DLTAY3
       STA DELTAY
       LDA FLAG3
       STA FLAG
       JSR ENDPTS
       STA YBOT
       JSR VLINE
       LDA XPLT+1
       CMP XMID+1
       BNE NEXTX1
       LDA XPLT
       CMP XMID
       BEQ CONT
NEXTX1 INC XPLT
       BNE SKIP3
       INC XPLT+1
SKIP3  JMP FCETLP
CONT   SEC
       LDA XPLT
       SBC XMIN
       STA XDIFF
       LDA DLTAX3
       BEQ FINI
       STA DELTAX
       LDA DLTAY3
       STA DELTAY
       LDA FLAG3
       STA FLAG
       LDA YMIN
       STA YBASE
       JSR ENDPTS
       STA YBOT
       SEC
       LDA XPLT
       SBC XMID
       STA XDIFF
       LDA DLTAX2
       BEQ FINI
       STA DELTAX
       LDA DLTAY2
       STA DELTAY
       LDA FLAG2
       STA FLAG
       LDA YMID
       STA YBASE
       JSR ENDPTS
       STA YTOP
       JSR VLINE
       LDA XPLT+1
       CMP XMAX+1
       BNE NEXTX2
       LDA XPLT
       CMP XMAX
       BEQ FINI
NEXTX2 INC XPLT
       BNE SKIP4
       INC XPLT+1
SKIP4  JMP CONT
FINI   LDA EDGES
       BEQ FINISH
       JSR OUTLN
       JSR SWAP23
       JSR OUTLN
       JSR SWAP12
       JSR SWAP23
       JSR SWAP12
       JSR OUTLN
FINISH RTS
.END
```

Listing 3. Assembler-Quelltext zum Zeichnen von Linien und Dreiecken (Schluß)


```
; PRIMITIVE SOLID SHAPE DRAWING
;
; RICHARD L. RYLANDER   11/7/84
;
; LOAD ARITHMETIC AND GRAPHIC UTILITIES FIRST
;
;****************************************
RAM=$036A
ORIGIN=$C5EA
;
MLPCND=$AC      ; MULTIPLICAND (S)
MLPLER=$AD      ; MULTIPLIER (S)
PROD=$AE        ; PRODUCT (D)
MULT=$C011      ; CALL FOR MULTIPLY
;
DVDND=$FD       ; DIVIDEND (D)
DVSOR=$FB       ; DIVISOR (D)
QUOT=$FD        ; QUOTIENT (D)
DIVIDE=$C025    ; CALL FOR DIVIDE
;
ARG=$AC         ; ARGUMENT (S)
SQR=$AE         ; SQUARE OF ARG (D)
SQUARE=$C004    ; CALL FOR SQUARE
;
RADCND=$AC      ; RADICAND (D)
ROOT=$033C      ; SQUARE ROOT (S)
SQRT=$C064      ; CALL FOR SQRT
;
RNDM=$C000      ; RANDOM NUMBER
RANDOM=$C0C8    ; CALL FOR RANDOM
; NOTE - A CALL TO 'RANDOM' LEAVES A RANDOM BYTE
; IN THE ACCUMULATOR
;
XPLT=$033F
YPLT=$0341
NORM=$C224
PLTSHD=$C20F
VALUE=$0344     ; FINAL NORMALIZED SHADE VALUE
HTORRN=$0346    ; SHADE FLAG, 1=HALFTONE
NOSCAL=$0347    ; SCALE FLAG, 1=NO SCALE
;
       *=RAM
XCENT  *=*+2    ; CENTER COORD
XREL   *=*+1    ; RELATIVE (TO CENTER)
XSHD   *=*+2    ; USED IN SHADE CALC
YCENT  *=*+1    ; CENTER COORD
YREL   *=*+1    ; RELATIVE (TO CENTER)
YSHD   *=*+2    ; USED IN SHADE CALC
ZREL   *=*+2    ; RELATIVE (TO CENTER)
ZWX    *=*+2    ; Z WITH X (+ OR -)
;
RADIUS *=*+2    ; LOCAL RADIUS OF SURFACE
TONE   *=*+2    ; USED IN SHADE CALC
TNTMP  *=*+2    ; USED IN SHADE CALC
;
CLIPL  *=*+1    ; LEFT CLIPPING BOUND
CLIPR  *=*+1    ; RIGHT CLIPPING BOUND
CLIPU  *=*+1    ; UP CLIPPING BOUND
CLIPD  *=*+1    ; DOWN CLIPPING BOUND
;
HEMI   *=*+1    ; PLOTTING HEMISPHERE
;
BAKLIT *=*+1    ; BACKLIT FLAG
HVFLAG *=*+1    ; HORIZONTAL/VERTICAL FLAG
TEMP   *=*+2    ; TEMPORARY STORAGE
CNTX   *=*+1     ; LOOP COUNTER
CNTY   *=*+1     ; LOOP COUNTER
MAX    *=*+1     ; LOOP LIMIT
;
HLEN   *=*+1     ; HALF-LENGTH OF CYLINDERS
RS     *=*+2     ; SQUARE OF TOROID RADIUS
RT     *=*+1     ; TOROID (RING) RADIUS
RC     *=*+1     ; CENTER RADIUS OF TOROID
RO     *=*+1     ; OUTER RADIUS OF TOROID
RI     *=*+1     ; INNER RADIUS OF TOROID
XSQR   *=*+2
XMAX   *=*+1
;
R0=HLEN
;
       *=ORIGIN
;****************************************
;
; DIVIDE WITH SINGLE PRECISION DIVISOR
; (USED OFTEN IN SHAPE ROUTINES)
;
SDIV   LDA #0
       STA DVSOR+1
       JMP DIVIDE
;
;****************************************
;
; CALCULATE SHADE VALUE (0-63) BY
; MULTIPLYING 'TONE' BY 26 THEN
; DIVIDE RESULT BY RADIUS OF SURFACE
;
GETVAL BIT TONE+1
       BPL CNTNU    ; IF 'TONE'<0, THEN
       LDA BAKLIT   ; MAKE VALUE 0 OR ABS(TONE)
       BNE NEGATE   ; DEPENDING ON BAKLIT FLAG
       STA VALUE
       RTS
NEGATE SEC
       LDA #$00
       SBC TONE
       STA TONE
CNTNU  LDA TONE
       STA MLPCND
       LDA #$1A
       STA MLPLER
       JSR MULT
       STA DVDND+1
       LDA PROD
       STA DVDND
       LDA RADIUS
       STA DVSOR
       JSR SDIV
       LDA QUOT
       STA VALUE
       RTS
;
;****************************************
;
; POINT PLOTTING BY QUADRANTS USING
; THE FOUR-FOLD SYMMETRY OF SIMPLE OBJECTS
;
; DEPENDING ON STATUS OF 'HVFLAG', EXCHANGE
; X AND Y COORDINATES TO ROTATE OBJECTS 90 DEG
; SINGLE SHAPE ROUTINE CAN THEN BE USED TO
; DRAW 'HORIZONTAL' OR 'VERTICAL' VERSIONS
; OF AN OBJECT
;
; THE FOLOWING IS A 'BASIC SUBROUTINE'
; EQUIVALENT TO EXPLAIN ITS OPERATION
;
; NOTE THAT LABELS ARE USED IN PLACE OF
; LINE NUMBERS
;
; 'PTPLOT' IF HVFLAG<0 THEN GOTO 'NOROT'
;          (STACK)=XREL:XREL=YREL:YREL=(STACK)
;          (STACK)=XSHD:XSHD=YSHD:YSHD=(STACK)
; 'NOROT'  GOSUB 'GETZ'
;          REM CALCULATE 2*Z FROM X,Y AND RADIUS
;          HEMI = 1
;          IF XREL>CLIPL THEN GOTO 'RHEMI'
;          ZWX=2*Z-XSHD
;          XPLT=XCENT-XREL:REM LEFT HEMISPHERE
; 'CHCLUP' IF YREL>CLIPU THEN GOTO 'DHEMI'
;          TONE=ZWX+YSHD
;          GOSUB 'GETVAL':REM NORMALIZE SHADE VAL
;          YPLT=YCENT+YREL
;          GOSUB 'PLTSHD':REM PLOT OR UNPLOT
;          REM   POINTS WEIGHTED BY SHADE VALUE
; 'DHEMI'  IF YREL>CLIPD THEN GOTO 'RHEMI'
;          TONE=ZWX-YSHD
;          GOSUB 'GETVAL'
;          YPLT=YCENT-YREL
;          GOSUB 'PLTSHD'
; 'RHEMI'  IF HEMI=0 THEN RETURN
;          HEMI=0
;          IF XREL>CLIPR THEN RETURN
;          ZWX=2*Z+XSHD
;          XPLT=XCENT+XREL
;          GOSUB 'CHCLUP'
;          RETURN
;
PTPLOT BIT HVFLAG
       BPL NOROT
       LDA XREL
       PHA
       PHA
       LDA YREL
       STA XREL
       PLA
       STA YREL
       LDA XSHD
       PHA
       PHA
       LDA YSHD
       STA XSHD
       PLA
       STA YSHD
       LDA XSHD+1
       PHA
       PHA
       LDA YSHD+1
       STA XSHD+1
       PLA
       STA YSHD+1
NOROT  JSR GETZ
PTPLT2 LDA #$01
       STA HEMI
       SEC
       LDA CLIPL  ; CHECK LEFT HEMISPHERE
       CMP XREL
       BCC RHEMI
       SEC
       LDA ROOT
       SBC XSHD
       STA ZWX
       LDA ROOT+1
       SBC XSHD+1
       STA ZWX+1
       SEC
       LDA XCENT
       SBC XREL
       STA XPLT
       LDA XCENT+1
       SBC #$00
       STA XPLT+1
CHCLUP SEC
       LDA CLIPU  ; CHECK FOR UP CLIPPING
       CMP YREL
       BCC DHEMI
       CLC
       LDA ZWX
       ADC YSHD
```

Listing 4. Assembler-Quelltext zum Zeichnen ausgefüllter Flächen, sogenannter Shapes

```
        STA TONE
        LDA ZWX+1
        ADC YSHD+1
        STA TONE+1
        JSR GETVAL
        CLC
        LDA YCENT
        ADC YREL
        STA YPLT
        JSR PLTSHD
;
DHEMI   SEC
        LDA CLIPD  ; CHECK FOR DOWN CLIPPING
        CMP YREL
        BCC RHEMI
        SEC
        LDA ZWX
        SBC YSHD
        STA TONE
        LDA ZWX+1
        SBC YSHD+1
        STA TONE+1
        JSR GETVAL
        SEC
        LDA YCENT
        SBC YREL
        STA YPLT
        JSR PLTSHD
;
RHEMI   LDA HEMI
        BEQ PLDONE
        DEC HEMI
        SEC
        LDA CLIPR  ; CHECK FOR RIGHT CLIPPING
        CMP XREL
        BCC PLDONE
        CLC
        LDA ROOT
        ADC XSHD
        STA ZWX
        LDA ROOT+1
        ADC XSHD+1
        STA ZWX+1
        CLC
        LDA XCENT
        ADC XREL
        STA XPLT
        LDA XCENT+1
        ADC #$00
        STA XPLT+1
        JMP CHCLUP
PLDONE  BIT HVFLAG
        BPL NORSTR
        LDA XSHD+1     ; RESTORE COORDS
        STA YSHD+1
        PLA
        STA XSHD+1
        LDA XSHD
        STA YSHD
        PLA
        STA XSHD
        LDA XREL
        STA YREL
        PLA
        STA XREL
NORSTR  RTS
;
;****************************************
;
; CALCULATE Z FROM LOCAL X,Y BY
; PYTHAGOREAN SUM
;
GETZ    LDA RADIUS
        STA ARG
        JSR SQUARE
        STA TNTMP+1
        LDA SQR
        STA TNTMP
        LDA XSHD
        STA ARG
        JSR SQUARE
        SEC
        LDA TNTMP
        SBC SQR
        STA TNTMP
        LDA TNTMP+1
        SBC SQR+1
        STA TNTMP+1
        LDA YSHD
        STA ARG
        JSR SQUARE
        SEC
        LDA TNTMP
        SBC SQR
        STA RADCND
        LDA TNTMP+1
        SBC SQR+1
        STA RADCND+1
        BMI ZEROOT
        JSR SQRT
        ASL ROOT
        ROL ROOT+1
        RTS
ZEROOT  LDA #$00
        STA ROOT
        STA ROOT+1
        RTS
;
;****************************************
;
; SET UP PARAMETERS FOR TOROIDS
;
; RT=(RO-RI)/2   RS=RT*RT   RC=RT+RI
;
TPARM   LDA RO
        SEC
        SBC RI
        LSR A
        STA RT
        STA RADIUS
        CLC
        ADC RI
        STA RC
        LDA RT
        STA ARG
        JSR SQUARE
        LDA SQR
        STA RS
        LDA SQR+1
        STA RS+1
        LDA #0
        STA CNTX
        RTS
;
;****************************************
;
; DRAW A SHADED SPHERE
;
; 'BASIC SUBROUTINE' EQUIVALENT
;
; 'SPHERE' FOR CNTX=0 TO RADIUS/SQR(2)
;          XREL=CNTX:XSHD=CNTX
;          FOR CNTY=CNTX TO SQR(RAD*RAD-CNTX*CNTX)
;          YREL=CNTY:YSHD=CNTY
;          HVFLAG=0
;          GOSUB 'PTPLOT'
;          REM EXCHANGE X & Y TO USE 8-FOLD SYM
;          HVFLAG=-128
;
;          GOSUB 'PTPLOT'
;          NEXT CNTY
;          NEXT CNTX
;          RETURN
;
;
SPHERE  LDA RADIUS
        STA ARG
        JSR SQUARE
        ASL SQR
        ROL SQR+1
        LDA SQR
        STA RADCND
        LDA SQR+1
        STA RADCND+1
        JSR SQRT
        LSR ROOT+1
        ROR ROOT
        LDA ROOT
        STA XMAX
        LDA #$00
        STA CNTX
        STA XSHD+1
        STA YSHD+1
        LDA RADIUS
        STA ARG
        JSR SQUARE
        STA TEMP+1
        LDA SQR
        STA TEMP
LOOPX   LDA CNTX
        STA CNTY
        STA ARG
        STA XREL
        STA XSHD
        JSR SQUARE
        SEC
        LDA TEMP
        SBC SQR
        STA RADCND
        LDA TEMP+1
        SBC SQR+1
        STA RADCND+1
        JSR SQRT
        LDA ROOT
        STA MAX
LOOPY   LDA CNTY
        STA YREL
        STA YSHD
        LDA #0
        STA HVFLAG
        JSR PTPLOT
        LDA #$80
        STA HVFLAG
        JSR PTPLOT
        LDA CNTY
        CMP MAX
        BEQ DONEY
        INC CNTY
        JMP LOOPY
DONEY   LDA CNTX
        CMP XMAX
        BEQ DONE
        INC CNTX
        JMP LOOPX
DONE    RTS
;
;****************************************
;
; DRAW SHADED CYLINDERS
;
; 'BASIC SUBROUTINE' EQUIVALENT
;
; 'CYLNDR' XSHD=0
;          FOR YREL=RADIUS TO 0
;          YSHD=YREL
;          FOR XREL=HLEN TO 0
;          GOSUB 'PTPLOT'
;          NEXT XREL
;          NEXT YREL
;          RETURN
;
CYLNDR  LDA #0
        STA XSHD
        STA XSHD+1
        STA YSHD+1
        LDA RADIUS
        STA YREL
CYLOOP  LDA HLEN
        STA XREL
        LDA YREL
        STA YSHD
CXLOOP  JSR PTPLOT
        DEC XREL
        BPL CXLOOP
        DEC YREL
        BPL CYLOOP
        RTS
;
;****************************************
;
; DRAW EDGE-VIEW TOROIDS
;
; 'BASIC SUBROUTINE' EQUIVALENT
;
; 'EDGTOR' GOSUB 'TPARM':REM SET UP RADII
;          FOR CNTX=0 TO RT
;          XREL=CNTX:XSHD=CNTX
;          R0=SQR(RT*RT-CNTX*CNTX)
;          FOR CNTY=0 TO R0+RC
;          YREL=CNTY
;          YSHD=(R0*CNTY)/(R0+RC)
;          GOSUB 'PTPLOT'
;          NEXT CNTY
;          NEXT CNTX
;          RETURN
;
EDGTOR  JSR TPARM
        LDA #$00
        STA XSHD+1
        STA YSHD+1
LOOPX4  LDA CNTX
        STA XREL
        STA XSHD
        STA ARG
        JSR SQUARE
        SEC
        LDA RS
        SBC SQR
        STA RADCND
        LDA RS+1
        SBC SQR+1
        STA RADCND+1
        JSR SQRT
        LDA ROOT
        STA R0
        CLC
        ADC RC
        STA MAX
        LDA #$00
        STA CNTY
LOOPY4  LDA CNTY
        STA YREL
        STA MLPLER
        LDA R0
        STA MLPCND
        JSR MULT
        STA DVDND+1
        LDA PROD
        STA DVDND
        LDA MAX
        STA DVSOR
        JSR SDIV
        LDA QUOT
        STA YSHD
        JSR PTPLOT
        LDA CNTY
        CMP MAX
        BEQ DONE4
        INC CNTY
        JMP LOOPY4
DONE4   LDA CNTX
        CMP RT
        BEQ DONEHT
        INC CNTX
        JMP LOOPX4
DONEHT  RTS
;
;****************************************
;
; DRAW A SHADED, TOP-VIEW TOROID
;
; 'BASIC SUBROUTINE' EQUIVALENT
;
; 'TOROID' GOSUB 'TPARM'
;          FOR CNTX=0 TO RO/SQR(2)
;          REM 8-FOLD SYMMETRY USED
;          XREL=CNTX
;          MAX=SQR(RO*RO-CNTX*CNTX)
;          IF CNTX>RI THEN GOTO 'GRTR'
;          CNTY=SQR(RI*RI-CNTY*CNTY)
;          GOTO 'LLPY1'
; 'GRTR'   CNTY=CNTX
; 'LLPY1'  YREL=CNTY
;          R0=SQR(CNTY*CNTY+CNTX*CNTX)
;          XSHD=CNTX-(CNTX*RC)/R0
;          YSHD=CNTY-(CNTY*RC)/R0
;          HVFLAG=0:GOSUB 'PTPLOT'
;          HVFLAG=-128:GOSUB 'PTPLOT'
;          IF CNTY=MAX THEN GOTO 'DDNY1'
;          CNTY=CNTY+1
;          GOTO 'LLPY1'
; 'DDNY1'  NEXT CNTX
;          RETURN
;
TOROID  JSR TPARM
        LDA RO
        STA ARG
        JSR SQUARE
        ASL SQR
        ROL SQR+1
        LDA SQR
        STA RADCND
        LDA SQR+1
        STA RADCND+1
        JSR SQRT
        LSR ROOT+1
        ROR ROOT
        LDA ROOT
        STA XMAX
LLPX1   LDA CNTX
        STA XREL
        STA ARG
        JSR SQUARE
        STA XSQR+1
        LDA SQR
        STA XSQR
        LDA RO
        STA ARG
        JSR SQUARE
        SEC
        LDA SQR
        SBC XSQR
        STA RADCND
        LDA SQR+1
        SBC XSQR+1
        STA RADCND+1
        JSR SQRT
        LDA ROOT
        STA MAX
        SEC
        LDA RI
        SBC CNTX
        BCC GRTR
        LDA RI
        STA ARG
        JSR SQUARE
        SEC
        LDA SQR
        SBC XSQR
        STA RADCND
        LDA SQR+1
        SBC XSQR+1
        STA RADCND+1
        JSR SQRT
        LDA ROOT
        STA CNTY
        JMP LLPY1
GRTR    LDA CNTX
        STA CNTY
LLPY1   LDA CNTY
        STA YREL
        STA ARG
        JSR SQUARE
        CLC
        LDA SQR
        ADC XSQR
        STA RADCND
        LDA SQR+1
        ADC XSQR+1
        STA RADCND+1
        JSR SQRT
        LDA ROOT
        STA R0
        STA DVSOR
        LDA CNTX
        STA MLPLER
        LDA RC
        STA MLPCND
        JSR MULT
        STA DVDND+1
        LDA PROD
        STA DVDND
        JSR SDIV
        SEC
        LDA CNTX
        SBC QUOT
        STA XSHD
        LDA #$00
        SBC QUOT+1
        STA XSHD+1
        LDA CNTY
        STA MLPLER
        LDA RC
        STA MLPCND
        JSR MULT
        STA DVDND+1
        LDA PROD
        STA DVDND
        LDA R0
        STA DVSOR
```

Listing 4. Assembler-Quelltext zum Zeichnen ausgefüllter Flächen, sogenannter Shapes (Fortsetzung)

```
        JSR SDIV
        SEC
        LDA CNTY
        SBC QUOT
        STA YSHD
        LDA #$00
        STA HVFLAG
        SBC QUOT+1
        STA YSHD+1
        JSR PTPLOT
        LDA #$80
        STA HVFLAG
        JSR PTPLOT
        LDA CNTY
        CMP MAX
        BEQ DDNY1
        INC CNTY
        JMP LLPY1
DDNY1   LDA CNTX
        CMP XMAX
        BEQ DUNTOR
        INC CNTX
        JMP LLPX1
DUNTOR RTS
;
;****************************************
;
; DRAW "INSIDE VIEW" TOROIDS
;
; 'BASIC SUBROUTINE' EQUIVALENT
;
; 'SPOOL'  GOSUB 'TPARM'
;          FOR CNTX=0 TO RT
;          XREL=CNTX:XSHD=CNTX
;          MAX=RC-SQR(RS-CNTX*CNTX)
;          FOR CNTY=0 TO MAX
;          YREL=CNTY
;          YSHD=(RC*CNTY/MAX)-CNTY
;          GOSUB 'PTPLOT'
;          NEXT CNTY
;          NEXT CNTX
;          RETURN
;
SPOOL   JSR TPARM
LLPX2   LDA CNTX
        STA XREL
        STA ARG
        SEC
        LDA #$00
        SBC CNTX
        STA XSHD
        LDA #$00
        SBC #$00
        STA XSHD+1
        JSR SQUARE
        SEC
        LDA RS
        SBC SQR
        STA RADCND
        LDA RS+1
        SBC SQR+1
        STA RADCND+1
        JSR SQRT
        SEC
        LDA RC
        SBC ROOT
        STA MAX
        LDA #$00
        STA CNTY
LLPY2   LDA CNTY
        STA YREL
        STA MLPLER
        LDA RC
        STA MLPCND
        JSR MULT
        STA DVDND+1
        LDA PROD
        STA DVDND
        LDA MAX
        STA DVSOR
        JSR SDIV
        LDA QUOT
        SEC
        SBC CNTY
        STA YSHD
        LDA QUOT+1
        SBC #$00
        STA YSHD+1
        JSR PTPLOT
        LDA CNTY
        CMP MAX
        BEQ DDNY2
        INC CNTY
        JMP LLPY2
DDNY2   LDA CNTX
        CMP RT
        BEQ DUNHSP
        INC CNTX
        JMP LLPX2
DUNHSP RTS
.END
```

Listing 4. Assembler-Quelltext zum Zeichnen ausgefüllter Flächen, sogenannter Shapes (Schluß)

```
; INTERFACE - EASY PARAMETER SETTING FOR SHAPE
; DRAWING ROUTINES FROM BASIC.
;
; RICHARD L. RYLANDER  11/23/84
;
;****************************************
ORIGIN=$CAC8
RAM   =$0393
;
; PARAMETER LOCATIONS FOR VARIOUS SHAPES
;
XCENT =$036A
YCENT =$036F
XPLOT =$033F
YPLOT =$0341
XMIN  =$034A
YMIN  =$034C
XMID  =$034D
YMID  =$034F
XMAX  =$0350
YMAX  =$0352
RADIUS =$0377
HLEN  =$0389
RI    =$038F
RO    =$038E
;
HVFLAG =$0383
VALUE =$0344
PLTFLG =$033E
;
DEFLAG =$FB
;
;****************************************
;
; FUNCTION LOCATIONS
;
GRFON =$C0E2       ; SWITCH TO GRAPHICS MODE
GRFOFF =$C103       ; RETURN TO TEXT DISPLAY
;
CLEARR=$C12C        ; CLEAR BITMAP
CLRBYT=$C135        ; CLEAR (FILL) BYTE
COLORR=$C118        ; LOAD COLOR MAP
COLBYT=$C119        ; COLOR BYTE
;
PLOTR =$C14B       ; POINT PLOT ROUTINE
LINER =$C2DB        ; DRAW A LINE
FACETR =$C4E1        ; DRAW A SHADED FACET
;
;****************************************
;
; SHADED SHAPE DRAWING ROUTINES
;
SPHERR=$C7C7        ; SPHERE
CYLNDR=$C864        ; CYLINDER
TORUSR=$C90F        ; TOP-VIEW TOROID
EDGTOR=$C88F        ; EDGE-VIEW TOROID
SPOOLR=$CA3B        ; INSIDE-VIEW TOROID
;
;****************************************
;
; BASIC ROM ROUTINES
;
CHKCOM=$AEFD        ; CHECK FOR COMMA
EVAEXP=$AD9E        ; EVALUATE EXPRESSION
FLTFIX=$B1AA        ; CONVERT TO FIXED
;
*=RAM
LINFAC *=*+1        ; LINE OR FACET FLAG
;
*=ORIGIN
;
;****************************************
;
; GET PARAMETERS FROM BASIC CALLING STATEMENT
; OF THE FORM:
; SYS(FNCTN),PARAM1,PARAM2,PARAM3AOPTU
; WHERE THE THIRD PARAMETER (FOR EXAMPLE)
; MAY BE OPTIONAL (A DEFAULT VALUE IS USED
; IF THE PARAMETER IS NOT SPECIFIED)
;
GETNUM JSR CHKCOM   ; LOOK FOR COMMA
       JSR EVAEXP   ; EVALUATE EXPRESSION
       JSR FLTFIX   ; CHANGE TO INTEGER WITH
; HIGH BYTE IN "A" AND LOW BYTE IN "Y"
       RTS
;
; CHECK FOR ADDITIONAL (OPTIONAL) PARAMETERS
;
PCHECK LDA #$2C     ; "," COMMA
       LDY #0
       STY DEFLAG
       CMP ($7A),Y
       BNE NOMORE   ; NO COMMA - USE DEFAULT
       JMP $0073
NOMORE LDY #$80
       STY DEFLAG
       RTS
;
; GET TWO ADDITIONAL PARAMETERS FOR TOROIDS
;
GETTWO JSR PCHECK
       BIT DEFLAG
       BMI DFAULT
       JSR EVAEXP
       JSR FLTFIX
       STY RI
       JSR GETNUM
       STY RO
DFAULT RTS
;
;****************************************
;
; SET CENTER COORDINATES
;
CENTER JSR GETNUM
       STY XCENT
       STA XCENT+1
       JSR GETNUM
       STY YCENT
       RTS
;
;****************************************
;
; CLEAR THE BITMAP, FILLING WITH (OPTIONAL)
; FILL VALUE SPECIFIED OR WITH (DEFAULT) "0"
;
CLEAR2 JSR PCHECK
       BIT DEFLAG
       BMI DEFCLR
       JSR EVAEXP
       JSR FLTFIX
       .BYTE $2C
DEFCLR LDY #0
       STY CLRBYT
       JMP CLEARR
;
;****************************************
;
; FILL COLOR MAP WITH (OPTIONAL) COLOR BYTE
; SPECIFIED OR WITH (DEFAULT) "$01"
; (BLACK DOTS ON WHITE BACKGROUND)
;
COLOR2 JSR PCHECK
       BIT DEFLAG
       BMI DEFCOL
       JSR EVAEXP
       JSR FLTFIX
       .BYTE $2C
DEFCOL LDY #$01
       STY COLBYT
       JMP COLORR
;
;****************************************
;
; PLOT OR UNPLOT POINTS
;
PLOT2  LDA #0
       .BYTE $2C
UNPLT2 LDA #$80
       STA PLTFLG
       JSR GETNUM
       STY XPLOT
       STA XPLOT+1
       JSR GETNUM
       STY YPLOT
       JMP PLOTR
;
;****************************************
;
; DRAW LINES BETWEEN (X1,Y1) AND (X2,Y2)
; OR SHADED FACETS BETWEEN THREE POINTS
; (X1,Y1), (X2,Y2) AND (X3,Y3)
;
LINE2  LDA #0
       .BYTE $2C
FACET2 LDA #$80
       STA LINFAC
       JSR GETNUM
       STY XMIN
       STA XMIN+1
       JSR GETNUM
       STY YMIN
       JSR GETNUM
       STY XMID
       STA XMID+1
       JSR GETNUM
       STY YMID
       BIT LINFAC
       BPL LDRAW
       JSR GETNUM
       STY XMAX
       STA XMAX+1
       JSR GETNUM
       STY YMAX
       JSR GETNUM
       STY VALUE
       JMP FACETR
LDRAW  JMP LINER
;
;****************************************
;
; DRAW A SPHERE CENTERED AT (XCENT,YCENT)
; DEFAULT RADIUS IS LAST VALUE USED
;
SPHER2 JSR CENTER
       JSR PCHECK
       BIT DEFLAG
       BMI SKIP1
       JSR EVAEXP
       JSR FLTFIX
       STY RADIUS
SKIP1  JMP SPHERR
;
;****************************************
;
; DRAW A TOP-VIEW TOROID AT (XCENT,YCENT)
; DEFAULT INNER AND OUTER RADII ARE LAST USED
;
TORUS2 JSR CENTER
       JSR GETTWO
       JMP TORUSR
;
;****************************************
;
; DRAW CYLINDERS WITH AXES HORIZONTAL OR
; VERTICAL.  DEFAULT RADIUS AND "HALF-LENGTH"
; ARE LAST VALUES USED.
;
VCYL2  LDA #$80
       .BYTE $2C
HCYL2  LDA #0
       STA HVFLAG
       JSR CENTER
       JSR PCHECK
       BIT DEFLAG
       BMI SKIP2
       JSR EVAEXP
       JSR FLTFIX
       STY RADIUS
       JSR GETNUM
       STY HLEN
SKIP2  JMP CYLNDR
;
;****************************************
;
; DRAW EDGE-VIEW TOROIDS WITH AXES HORIZONTAL
; OR VERTICAL
; INNNER AND OUTER RADII ARE OPTIONAL
;
VTOR2  LDA #$80
       .BYTE $2C
HTOR2  LDA #0
       STA HVFLAG
       JSR CENTER
       JSR GETTWO
       JMP EDGTOR
;
;****************************************
;
; DRAW INSIDE-VIEW TOROIDS, "SPOOLS",
; WITH AXES HORIZONTAL OR VERTICAL
; INNER AND OUTER RADII ARE OPTIONAL
;
VSPL2  LDA #$80
       .BYTE $2C
HSPL2  LDA #0
       STA HVFLAG
       JSR CENTER
       JSR GETTWO
       JMP SPOOLR
.END
```

Listing 5. Assembler-Quelltext des »Basic-Interfaces« zum einfachen Ansprechen aller Assembler-Routinen per SYS-Befehl

```
; KEYSORT - RELOCATABLE BUBBLE SORT USING KEY ARRAY
; POINTING TO INTEGER ARRAY
;
;  RICHARD L. RYLANDER    1/12/85
;
ORIGIN=$CF59  ; 53081. (FOLLOWING DOS 5.1)
;
KB     = $FB  ; 251. POINTER TO KEY ARRAY
ZB     = $FD  ; 253. POINTER TO DATA ARRAY
MAX    = $8C  ; 140. POKE WITH MAX ARRAY INDEX
TOP    = $AC
TOPDIS = $AD
FLAG   = $AE
NXTFLG = $61
```

Listing 6. Assembler-Quelltext der schnellen Bubble-Sort-Routine

```
CRRNT  = $62
REPEAT = $64
;
       *=ORIGIN
;
INIT   LDY #$FF      ; INITIALIZE KEY ARRAY
INLOOP INY
       TYA
       STA (KB),Y
       CMP MAX
       BNE INLOOP
;
SORT   STA TOPDIS    ; 'A' HOLDS 'MAX'
LOOP1  LDA TOPDIS
       STA TOP
       LDX #0
       STX NXTFLG
       STX FLAG
LOOP2  STX REPEAT
;
; GET BOTH BYTES OF INTEGER POINTED TO BY
; 'KEY' ELEMENT.  RETURN WITH MSB ON STACK
; AND LSB IN THE ACCUMULATOR
;
GETINT TXA
       TAY
       LDA (KB),Y
       ASL A
       BCC LOAD
       DEC NXTFLG
       INC ZB+1
LOAD   TAY
       LDA (ZB),Y
       PHA
       INY
       LDA (ZB),Y
       BIT NXTFLG
       BPL NODEC
       INC NXTFLG
       DEC ZB+1
NODEC  CPX REPEAT
       BNE ORDER
       STA CRRNT
       PLA
       STA CRRNT+1
       INX
       BNE GETINT
;
; COMPARE INTEGERS OBTAINED THROUGH KEY ARRAY
; IF 'CURRENT' >= 'NEXT' THEN SWAP KEY
; ELEMENTS, ELSE CONTINUE
;
ORDER  CMP CRRNT
       PLA
       SBC CRRNT+1
       BVC TEST
       EOR #$80
TEST   BPL NOSWAP
SWAP   TXA
       TAY
       STX TOPDIS
       LDA (KB),Y
       PHA
       DEY
       LDA (KB),Y
       INY
       STA (KB),Y
       PLA
       DEY
       STA (KB),Y
       INC FLAG
NOSWAP CPX TOP
       BNE LOOP2
       LDA FLAG
       BNE LOOP1
;
; UNPACK THE BYTE ELEMENTS OF THE 'KEY' ARRAY
; INTO BASIC'S NORMAL 2-BYTE INTEGER FORMAT
;
UNPACK LDX MAX
       INX
PKLOOP DEX
       TXA
       TAY
       LDA (KB),Y
       PHA
       TXA
       ASL A      ; MOVE TO 2*I-1
       ORA #1
       BCC STORE
       INC NXTFLG
       INC KB+1
STORE  TAY
       PLA
       STA (KB),Y
       LDA #0
       DEY
       STA (KB),Y
       LDA NXTFLG
       BEQ OK
       DEC NXTFLG
       DEC KB+1
OK     TXA
       BNE PKLOOP
DONE   RTS
.END
```

Listing 6. Assembler-Quelltext der schnellen Bubble-Sort-Routine (Schluß)

```
;    "WRITE"    RICHARD L. RYLANDER
;               12/30/84
; REVISED  1/19/85  - ORIGIN MOVED TO $CFE5 (53221.)
;
;  PUT TEXT CHARACTERS ON GRAPHIC SCREEN
;  (UNDER BASIC ROM) IN VARIOUS STYLES
;
*=$CFE5  ; PUT CODE AFTER DOS 5.1
WRITE  LDA $01      ; SWITCH OUT BASIC ROM
       AND #$FE
       STA $01
       LDY #7
LOOP   LDA ($FD),Y ; READ CHARACTER BYTE
       AND ($FB),Y ; MODIFY W/SCREEN BYTE
       STA ($FB),Y ; STORE IN SCREEN
;
; POKE NEW LOGICAL OPERATOR TO REPLACE
; 'AND' (53231.) FOR DIFFERENT STYLES
; ORA=17.  BIT (NOP)=36.  AND=49.  EOR=81.
;
       DEY
       BPL LOOP
       LDA $01      ; RESTORE BASIC ROM
       ORA #1
       STA $01
       RTS
.END
```

Listing 7. Assembler-Quelltext der Routine zum Beschreiben des HiRes-Bildschirmes unterhalb des Basic-ROMs

```
 programm : graphics      c000 cbfc
------------------------------------
c000 : ff 55 00 00 a5 ac 10 07  b8
c008 : 38 a9 00 e5 ac 85 ac 85  86
c010 : ad a9 00 a2 08 46 ad 90  71
c018 : 03 18 65 ac 6a 66 ae ca  40
c020 : d0 f3 85 af 60 a9 00 85  9f
c028 : b4 85 b5 a2 10 26 fd 26  d7
c030 : fe 26 b4 26 b5 38 a5 b4  50
c038 : e5 fb a8 a5 b5 e5 fc 90  99
c040 : 04 84 b4 85 b5 ca d0 e5  25
c048 : 26 fd 26 fe 06 b4 26 b5  e0
c050 : b0 0b 38 a5 fb e5 b4 a5  56
c058 : fc e5 b5 b0 06 e6 fd d0  fb
c060 : 02 e6 fe 60 a2 08 a9 00  b2
c068 : 8d 3c 03 8d 3d 03 85 fb  80
c070 : 85 fc 0e 3c 03 2e 3d 03  1b
c078 : ee 3c 03 d0 03 ee 3d 03  02
c080 : 06 ac 26 ad 26 fb 26 fc  f0
c088 : 06 ac 26 ad 26 fb 26 fc  f8
c090 : 38 a5 fb ed 3c 03 a8 a5  21
c098 : fc ed 3d 03 90 12 85 fc  e4
c0a0 : 84 fb ee 3c 03 d0 03 ee  06
c0a8 : 3d 03 ca d0 c5 4c c1 c0  7b
c0b0 : 38 ad 3c 03 e9 01 8d 3c  84
c0b8 : 03 b0 03 ce 3d 03 ca d0  67
c0c0 : b1 6e 3d 03 6e 3c 03 60  ee
c0c8 : ad 00 c0 8d 02 c0 4d 01  b4
c0d0 : c0 2e 03 c0 6a 6e 03 c0  28
c0d8 : 8d 00 c0 ad 02 c0 8d 01  a9
c0e0 : c0 60 ad 11 d0 09 20 8d  4f
c0e8 : 11 d0 ad 00 dd 8d 42 03  26
c0f0 : 29 fc 09 01 8d 00 dd ad  a5
c0f8 : 18 d0 8d 43 03 a9 19 8d  41
c100 : 18 d0 60 ad 11 d0 29 df  4a
c108 : 8d 11 d0 ad 42 03 8d 00  7a
c110 : dd ad 43 03 8d 18 d0 60  93
c118 : a9 01 a2 00 9d 00 84 9d  12
c120 : 00 85 9d 00 86 9d 00 87  af
c128 : ca d0 f1 60 a9 a0 85 fc  92
c130 : a0 00 84 fb a9 00 a2 20  d6
c138 : 91 fb c8 d0 fb e6 fc ca  93
c140 : d0 f6 60 a9 00 2c a9 80  e2
c148 : 8d 3e 03 a5 01 29 fe 85  ca
c150 : 01 38 a9 c7 ed 41 03 aa  1b
c158 : 4a 4a 4a a8 b9 ab c1 85  7a
c160 : fb b9 c4 c1 85 fc 8a 29  5e
c168 : 07 18 65 fb 85 fb ad 3f  c1
c170 : 03 29 f8 65 fb 85 fb ad  2a
c178 : 40 03 65 fc 85 fc a9 a0  5b
c180 : 65 fc 85 fc ad 3f 03 29  97
c188 : 07 49 07 aa a9 01 ca 30  79
c190 : 03 0a d0 fa a0 00 2c 3e  63
c198 : 03 10 05 49 ff 31 fb 2c  e0
c1a0 : 11 fb 91 fb a5 01 09 01  1b
c1a8 : 85 01 60 00 40 80 c0 00  d1
c1b0 : 40 80 c0 00 40 80 c0 00  6c
c1b8 : 40 80 c0 00 40 80 c0 00  74
c1c0 : 40 80 c0 00 00 01 02 03  87
c1c8 : 05 06 07 08 0a 0b 0c 0d  d7
c1d0 : 0f 10 11 12 14 15 16 17  de
c1d8 : 19 1a 1b 1c 1e ad 3f 03  9b
c1e0 : 29 07 8d 48 03 ad 41 03  a2
c1e8 : 29 07 0a 0a 0a 0d 48 03  89
c1f0 : aa bd 2f c2 cd 44 03 10  c8
c1f8 : 03 4c 46 c1 4c 43 c1 20  11
c200 : c8 c0 4a 4a cd 44 03 10  2f
c208 : 03 4c 46 c1 4c 43 c1 ad  3c
c210 : 47 03 f0 10 ac 41 03 c8  89
c218 : 84 ad a9 d5 85 ac 20 11  f8
c220 : c0 8d 41 03 ad 46 03 f0  53
c228 : 03 4c dd c1 4c ff c1 00  cd
c230 : 08 35 3d 02 0a 37 3f 10  da
c238 : 18 25 2d 12 1a 27 2f 31  6a
c240 : 39 04 0c 33 3b 06 0e 21  43
c248 : 29 14 1c 23 2b 16 1e 03  c9
c250 : 0b 36 3e 01 09 34 3c 13  6f
c258 : 1b 26 2e 11 19 24 2c 32  fc
c260 : 3a 07 0f 30 38 05 0d 22  0c
c268 : 2a 17 1f 20 28 15 1d a0  cb
c270 : 06 a9 d5 85 ac b9 4c 03  41
c278 : 85 ad 20 11 c0 99 4c 03  0e
c280 : 88 88 88 10 f0 60 a0 02  09
c288 : b9 4a 03 48 b9 4d 03 99  75
c290 : 4a 03 68 99 4d 03 88 10  d8
c298 : ef 60 a0 02 b9 4d 03 48  c2
c2a0 : b9 50 03 99 4d 03 68 99  37
c2a8 : 50 03 88 10 ef 60 a2 02  2e
c2b0 : 38 ad 4d 03 ed 4a 03 ad  0b
c2b8 : 4e 03 ed 4b 03 b0 03 20  6f
c2c0 : 86 c2 ca f0 15 38 ad 50  e2
c2c8 : 03 ed 4d 03 ad 51 03 ed  c3
c2d0 : 4e 03 b0 dc 20 9a c2 4c  e2
c2d8 : b0 c2 60 a9 02 8d 51 03  0e
c2e0 : 20 ae c2 ad 47 03 f0 03  14
c2e8 : 20 6f c2 20 6f c4 ad 4a  dd
c2f0 : 03 8d 3f 03 ad 4b 03 8d  46
c2f8 : 40 03 ad 4c 03 8d 41 03  56
c300 : ad 57 03 d0 7d 38 ad 56  31
c308 : 03 ed 5b 03 b0 74 ad 5b  55
c310 : 03 8d 65 03 8d 68 03 4e  58
c318 : 65 03 38 ad 56 03 ed 65  c3
c320 : 03 8d 65 03 ad 57 03 e9  19
c328 : 00 8d 66 03 ee 68 03 ad  82
c330 : 67 03 d0 06 20 43 c1 4c  c9
c338 : 3d c3 20 46 c1 ad 60 03  39
c340 : d0 05 ee 41 03 d0 03 ce  d7
c348 : 41 03 2c 66 03 30 1a ee  db
c350 : 3f 03 d0 03 ee 40 03 38  13
c358 : ad 65 03 ed 5b 03 8d 65  05
c360 : 03 ad 66 03 e9 00 8d 66  d5
c368 : 03 18 ad 65 03 6d 56 03  8a
c370 : 8d 65 03 ad 66 03 6d 57  09
c378 : 03 8d 66 03 ce 68 03 d0  1a
c380 : ae 60 ad 56 03 8d 65 03  cd
c388 : 8d 68 03 ad 57 03 8d 66  50
c390 : 03 8d 69 03 4e 66 03 6e  16
c398 : 65 03 38 ad 5b 03 ed 65  93
c3a0 : 03 8d 65 03 a9 00 ed 66  43
c3a8 : 03 8d 66 03 ad 67 03 d0  30
c3b0 : 06 20 43 c1 4c ba c3 20  b9
c3b8 : 46 c1 ee 3f 03 d0 03 ee  23
c3c0 : 40 03 2c 66 03 30 20 ad  e7
c3c8 : 60 03 d0 05 ee 41 03 d0  25
c3d0 : 03 ce 41 03 38 ad 65 03  78
c3d8 : ed 56 03 8d 65 03 ad 66  55
c3e0 : 03 ed 57 03 8d 66 03 18  58
c3e8 : ad 65 03 6d 5b 03 8d 65  85
c3f0 : 03 ad 66 03 69 00 8d 66  5d
c3f8 : 03 38 ad 68 03 e9 01 8d  2e
c400 : 68 03 b0 03 ce 69 03 2c  13
c408 : 69 03 10 a0 60 38 ad 53  30
c410 : 03 ed 54 03 b0 0e ad 53  58
c418 : 03 48 ad 54 03 8d 53 03  25
c420 : 68 8d 54 03 ad 53 03 8d  61
c428 : 41 03 20 24 c2 ad 53 03  64
c430 : cd 54 03 f0 06 ce 53 03  30
c438 : 4c 24 c4 60 ad 5f 03 85  c0
c440 : ac ad 5e 03 85 ad 20 11  23
c448 : c0 85 fe a5 ae 85 fd a9  a1
c450 : 00 85 fc ad 5a 03 85 fb  d3
c458 : 20 25 c0 ad 63 03 d0 08  92
c460 : 18 ad 55 03 65 fd 90 06  99
c468 : 38 ad 55 03 e5 fd 60 38  6d
c470 : ad 4d 03 ed 4a 03 8d 56  e2
c478 : 03 ad 4e 03 ed 4b 03 8d  a6
c480 : 57 03 38 ad 50 03 ed 4d  8c
c488 : 03 8d 58 03 38 ad 50 03  01
c490 : ed 4a 03 8d 59 03 a9 00  69
c498 : 8d 60 03 8d 61 03 8d 62  f1
c4a0 : 03 38 ad 4f 03 ed 4c 03  eb
c4a8 : b0 09 ee 60 03 ad 4c 03  79
c4b0 : ed 4f 03 8d 5b 03 38 ad  c1
c4b8 : 52 03 ed 4f 03 b0 09 ee  a9
c4c0 : 61 03 ad 4f 03 ed 52 03  e7
c4c8 : 8d 5c 03 38 ad 52 03 ed  a1
c4d0 : 4c 03 b0 09 ee 62 03 ad  54
c4d8 : 4c 03 ed 52 03 8d 5d 03  84
c4e0 : 60 20 ae c2 ad 47 03 f0  57
c4e8 : 03 20 6f c2 20 6f c4 ad  1b
c4f0 : 4a 03 8d 3f 03 ad 4b 03  d8
c4f8 : 8d 40 03 38 ad 3f 03 ed  2a
c500 : 4a 03 8d 5f 03 ad 56 03  18
c508 : f0 53 8d 5a 03 ad 5b 03  62
c510 : 8d 5e 03 ad 60 03 8d 63  5e
c518 : 03 ad 4c 03 8d 55 03 20  35
c520 : 3c c4 8d 53 03 ad 59 03  95
c528 : f0 33 8d 5a 03 ad 5d 03  7a
c530 : 8d 5e 03 ad 62 03 8d 63  9e
c538 : 03 20 3c c4 8d 54 03 20  bb
c540 : 0d c4 ad 40 03 cd 4e 03  01
c548 : d0 08 ad 3f 03 cd 4d 03  49
c550 : f0 0b ee 3f 03 d0 03 ee  0a
c558 : 40 03 4c fb c4 38 ad 3f  f0
c560 : 03 ed 4a 03 8d 5f 03 ad  88
c568 : 59 03 f0 63 8d 5a 03 ad  fe
c570 : 5d 03 8d 5e 03 ad 62 03  ab
c578 : 8d 63 03 ad 4c 03 8d 55  eb
c580 : 03 20 3c c4 8d 54 03 38  33
c588 : ad 3f 03 ed 4d 03 8d 5f  35
c590 : 03 ad 58 03 f0 39 8d 5a  a4
c598 : 03 ad 5c 03 8d 5e 03 ad  1d
c5a0 : 61 03 8d 63 03 ad 4f 03  33
c5a8 : 8d 55 03 20 3c c4 8d 53  6b
c5b0 : 03 20 0d c4 ad 40 03 cd  24
c5b8 : 51 03 d0 08 ad 3f 03 cd  3c
c5c0 : 50 03 f0 0b ee 3f 03 d0  c6
c5c8 : 03 ee 40 03 4c 5d c5 ad  d5
c5d0 : 64 03 f0 15 20 eb c2 20  41
c5d8 : 9a c2 20 eb c2 20 86 c2  26
c5e0 : 20 9a c2 20 86 c2 20 eb  d9
c5e8 : c2 60 a9 00 85 fc 4c 25  00
c5f0 : c0 2c 7a 03 10 12 ad 82  13
c5f8 : 03 d0 04 8d 44 03 60 38  64
c600 : a9 00 ed 79 03 8d 79 03  dc
c608 : ad 79 03 85 ac a9 1a 85  6f
c610 : ad 20 11 c0 85 fe a5 ae  6e
```

Listing 8. MSE-Listing der Assembler-Routinen aus Listing 1 bis 5. Bitte mit dem MSE (Seite 8) eingeben.

(Seite 8)

```
c618 : 85 fd ad 77 03 85 fb 20   83
c620 : ea c5 a5 fd 8d 44 03 60   de
c628 : 2c 83 03 10 2d ad 6c 03   d1
c630 : 48 48 ad 70 03 8d 6c 03   6a
c638 : 68 8d 70 03 ad 6d 03 48   c6
c640 : 48 ad 71 03 8d 6d 03 68   3d
c648 : 8d 71 03 ad 6e 03 48 48   b5
c650 : ad 72 03 8d 6e 03 68 8d   65
c658 : 72 03 20 45 c7 a9 01 8d   e5
c660 : 81 03 38 ad 7d 03 cd 6c   27
c668 : 03 90 7d 38 ad 3c 03 ed   be
c670 : 6d 03 8d 75 03 ad 3d 03   09
c678 : ed 6e 03 8d 76 03 38 ad   cb
c680 : 6a 03 ed 6c 03 8d 3f 03   14
c688 : ad 6b 03 e9 00 8d 40 03   5c
c690 : 38 ad 7f 03 cd 70 03 90   6d
c698 : 23 18 ad 75 03 6d 71 03   49
c6a0 : 8d 79 03 ad 76 03 6d 72   7a
c6a8 : 03 8d 7a 03 20 f1 c5 18   4a
c6b0 : ad 6f 03 6d 70 03 8d 41   5b
c6b8 : 03 20 0f c2 38 ad 80 03   e0
c6c0 : cd 70 03 90 23 38 ad 75   2e
c6c8 : 03 ed 71 03 8d 79 03 ad   8b
c6d0 : 76 03 ed 72 03 8d 7a 03   1e
c6d8 : 20 f1 c5 38 ad 6f 03 ed   a7
c6e0 : 70 03 8d 41 03 20 0f c2   50
c6e8 : ad 81 03 f0 34 ce 81 03   fa
c6f0 : 38 ad 7e 03 cd 6c 03 90   6c
c6f8 : 28 18 ad 3c 03 6d 6d 03   77
c700 : 8d 75 03 ad 3d 03 6d 6e   3d
c708 : 03 8d 76 03 18 ad 6a 03   6e
c710 : 6d 6c 03 8d 3f 03 ad 6b   bf
c718 : 03 69 00 8d 40 03 4c 90   f0
c720 : c6 2c 83 03 10 1e ad 6e   c3
c728 : 03 8d 72 03 68 8d 6e 03   a2
c730 : ad 6d 03 8d 71 03 68 8d   f2
c738 : 6d 03 ad 6c 03 8d 70 03   84
c740 : 68 8d 6c 03 60 ad 77 03   42
c748 : 85 ac 20 04 c0 8d 7c 03   1c
c750 : a5 ae 8d 7b 03 ad 6d 03   78
c758 : 85 ac 20 04 c0 38 ad 7b   37
c760 : 03 e5 ae 8d 7b 03 ad 7c   33
c768 : 03 e5 af 8d 7c 03 ad 71   75
c770 : 03 85 ac 20 04 c0 38 ad   e7
c778 : 7b 03 e5 ae 85 ac ad 7c   31
c780 : 03 e5 af 85 ad 30 0a 20   d7
c788 : 64 c0 0e 3c 03 2e 3d 03   f4
c790 : 60 a9 00 8d 3c 03 8d 3d   03
c798 : 03 60 ad 8e 03 38 ed 8f   d1
c7a0 : 03 4a 8d 8c 03 8d 77 03   3e
c7a8 : 18 6d 8f 03 8d 8d 03 ad   68
c7b0 : 8c 03 85 ac 20 04 c0 a5   25
c7b8 : ae 8d 8a 03 a5 af 8d 8b   55
c7c0 : 03 a9 00 8d 86 03 60 ad   a7
c7c8 : 77 03 85 ac 20 04 c0 06   e9
c7d0 : ae 26 af a5 ae 85 ac a5   47
c7d8 : af 85 ad 20 64 c0 4e 3d   b9
c7e0 : 03 6e 3c 03 ad 3c 03 8d   6e
c7e8 : 92 03 a9 00 8d 86 03 8d   9a
c7f0 : 6e 03 8d 72 03 ad 77 03   13
c7f8 : 85 ac 20 04 c0 8d 85 03   f1
c800 : a5 ae 8d 84 03 ad 86 03   ae
c808 : 8d 87 03 85 ac 8d 6c 03   b9
c810 : 8d 6d 03 20 04 c0 38 ad   9b
c818 : 84 03 e5 ae 85 ac ad 85   ec
c820 : 03 e5 af 85 ad 20 64 c0   a1
c828 : ad 3c 03 8d 88 03 ad 87   cc
c830 : 03 8d 70 03 8d 71 03 a9   3a
c838 : 00 8d 83 03 20 28 c6 a9   f2
c840 : 80 8d 83 03 20 28 c6 ad   82
c848 : 87 03 cd 88 03 f0 06 ee   83
c850 : 87 03 4c 2e c8 ad 86 03   4c
c858 : cd 92 03 f0 06 ee 86 03   45
c860 : 4c 05 c8 60 a9 00 8d 6d   19
c868 : 03 8d 6e 03 8d 72 03 ad   02
c870 : 77 03 8d 70 03 ad 89 03   a4
c878 : 8d 6c 03 ad 70 03 8d 71   ea
c880 : 03 20 28 c6 ce 6c 03 10   f3
c888 : f8 ce 70 03 10 e7 60 20   66
c890 : 9a c7 a9 00 8d 6e 03 8d   ec
c898 : 72 03 ad 86 03 8d 6c 03   1c
c8a0 : 8d 6d 03 85 ac 20 04 c0   b3
c8a8 : 38 ad 8a 03 e5 ae 85 ac   fd
c8b0 : ad 8b 03 e5 af 85 ad 20   be
c8b8 : 64 c0 ad 3c 03 8d 89 03   38
c8c0 : 18 6d 8d 03 8d 88 03 a9   cf
c8c8 : 00 8d 87 03 ad 87 03 8d   0f
c8d0 : 70 03 85 ad ad 89 03 85   17
c8d8 : ac 20 11 c0 85 fe a5 ae   35
c8e0 : 85 fd ad 88 03 85 fb 20   6d
c8e8 : ea c5 a5 fd 8d 71 03 20   8f
c8f0 : 28 c6 ad 87 03 cd 88 03   9e
c8f8 : f0 06 ee 87 03 4c cc c8   ef
c900 : ad 86 03 cd 8c 03 f0 06   1c
c908 : ee 86 03 4c 9a c8 60 20   35
c910 : 9a c7 ad 8e 03 85 ac 20   1a
c918 : 04 c0 06 ae 26 af a5 ae   a8
c920 : 85 ac a5 af 85 ad 20 64   6a
c928 : c0 4e 3d 03 6e 3c 03 ad   ef
c930 : 3c 03 8d 92 03 ad 86 03   61
c938 : 8d 6c 03 85 ac 20 04 c0   ca
c940 : 8d 91 03 a5 ae 8d 90 03   ab
c948 : ad 8e 03 85 ac 20 04 c0   0b
c950 : 38 a5 ae ed 90 03 85 ac   55
c958 : a5 af ed 91 03 85 ad 20   d6
c960 : 64 c0 ad 3c 03 8d 88 03   dc
c968 : 38 ad 8f 03 ed 86 03 90   fb
c970 : 23 ad 8f 03 85 ac 20 04   f4
c978 : c0 38 a5 ae ed 90 03 85   0e
c980 : ac a5 af ed 91 03 85 ad   4b
c988 : 20 64 c0 ad 3c 03 8d 87   e1
c990 : 03 4c 9a c9 ad 86 03 8d   cf
c998 : 87 03 ad 87 03 8d 70 03   61
c9a0 : 85 ac 20 04 c0 18 a5 ae   c5
c9a8 : 6d 90 03 85 ac a5 af 6d   60
c9b0 : 91 03 85 ad 20 64 c0 ad   5d
c9b8 : 3c 03 8d 89 03 85 fb ad   b2
c9c0 : 86 03 85 ad ad 8d 03 85   3d
c9c8 : ac 20 11 c0 85 fe a5 ae   25
c9d0 : 85 fd 20 ea c5 38 ad 86   9b
c9d8 : 03 e5 fd 8d 6d 03 a9 00   94
c9e0 : e5 fe 8d 6e 03 ad 87 03   37
c9e8 : 85 ad ad 8d 03 85 ac 20   b0
c9f0 : 11 c0 85 fe a5 ae 85 fd   84
c9f8 : ad 89 03 85 fb 20 ea c5   d3
ca00 : 38 ad 87 03 e5 fd 8d [illegible]   b8
ca08 : 03 a9 00 8d 83 03 e5 fe   77
ca10 : 8d 72 03 20 28 c6 a9 80   fc
ca18 : 8d 83 03 20 28 c6 ad 87   aa
ca20 : 03 cd 88 03 f0 06 ee 87   96
ca28 : 03 4c 9a c9 ad 86 03 cd   e8
ca30 : 92 03 f0 06 ee 86 03 4c   08
ca38 : 35 c9 60 20 9a c7 ad 86   1a
ca40 : 03 8d 6c 03 85 ac 38 a9   77
ca48 : 00 ed 86 03 8d 6d 03 a9   e4
ca50 : 00 e9 00 8d 6e 03 20 04   7e
ca58 : c0 38 ad 8a 03 e5 ae 85   16
ca60 : ac ad 8b 03 e5 af 85 ad   73
ca68 : 20 64 c0 38 ad 8d 03 ed   21
ca70 : 3c 03 8d 88 03 a9 00 8d   3b
ca78 : 87 03 ad 87 03 8d 70 03   41
ca80 : 85 ad ad 8d 03 85 ac 20   48
ca88 : 11 c0 85 fe a5 ae 85 fd   1c
ca90 : ad 88 03 85 fb 20 ea c5   eb
ca98 : a5 fd 38 ed 87 03 8d 71   b1
caa0 : 03 a5 fe e9 00 8d 72 03   af
caa8 : 20 28 c6 ad 87 03 cd 88   1c
cab0 : 03 f0 06 ee 87 03 4c 7a   41
cab8 : ca ad 86 03 cd 8c 03 f0   8a
cac0 : 06 ee 86 03 4c 3e ca 60   e2
cac8 : 20 fd ae 20 9e ad 20 aa   c4
cad0 : b1 60 a9 2c a0 00 84 fb   b5
cad8 : d1 7a d0 03 4c 73 00 a0   1c
cae0 : 80 84 fb 60 20 d2 ca 24   b9
cae8 : fb 30 0f 20 9e ad 20 aa   f0
caf0 : b1 8c 8f 03 20 c8 ca 8c   b8
caf8 : 8e 03 60 20 c8 ca 8c 6a   0e
cb00 : 03 8d 6b 03 20 c8 ca 8c   92
cb08 : 6f 03 60 20 d2 ca 24 fb   21
cb10 : 30 07 20 9e ad 20 aa b1   8a
cb18 : 2c a0 00 8c 35 c1 4c 2c   11
cb20 : c1 20 d2 ca 24 fb 30 07   f0
cb28 : 20 9e ad 20 aa b1 2c a0   31
cb30 : 01 8c 19 c1 4c 18 c1 a9   d6
cb38 : 00 2c a9 80 8d 3e 03 20   e0
cb40 : c8 ca 8c 3f 03 8d 40 03   1c
cb48 : 20 c8 ca 8c 41 03 4c 4b   04
cb50 : c1 a9 00 2c a9 80 8d 93   67
cb58 : 03 20 c8 ca 8c 4a 03 8d   39
cb60 : 4b 03 20 c8 ca 8c 4c 03   96
cb68 : 20 c8 ca 8c 4d 03 8d 4e   f0
cb70 : 03 20 c8 ca 8c 4f 03 2c   b6
cb78 : 93 03 10 18 20 c8 ca 8c   21
cb80 : 50 03 8d 51 03 20 c8 ca   c9
cb88 : 8c 52 03 20 c8 ca 8c 44   a0
cb90 : 03 4c e1 c4 4c db c2 20   b9
cb98 : fb ca 20 d2 ca 24 fb 30   79
cba0 : 09 20 9e ad 20 aa b1 8c   4e
cba8 : 77 03 4c c7 c7 20 fb ca   b0
cbb0 : 20 e4 ca 4c 0f c9 a9 80   65
cbb8 : 2c a9 00 8d 83 03 20 fb   33
cbc0 : ca 20 d2 ca 24 fb 30 0f   a9
cbc8 : 20 9e ad 20 aa b1 8c 77   00
cbd0 : 03 20 c8 ca 8c 89 03 4c   28
cbd8 : 64 c8 a9 80 2c a9 00 8d   46
cbe0 : 83 03 20 fb ca 20 e4 ca   43
cbe8 : 4c 8f c8 a9 80 2c a9 00   73
cbf0 : 8d 83 03 20 fb ca 20 e4   64
cbf8 : ca 4c 3b ca ff 00 ff 00   10
```

Listing 8. MSE-Listing der Assembler-Routinen aus Listing 1 bis 5 (Schluß)

```
programm : sort/write          cf59 cffd
-----------------------------------------
cf59 : a0 ff c8 98 91 fb c5 8c   67
cf61 : d0 f8 85 ad a5 ad 85 ac   fc
cf69 : a2 00 86 61 86 ae 86 64   9a
cf71 : 8a a8 b1 fb 0a 90 04 c6   fe
cf79 : 61 e6 fe a8 b1 fd 48 c8   e0
cf81 : b1 fd 24 61 10 04 e6 61   e6
cf89 : c6 fe e4 64 d0 08 85 62   bc
cf91 : 68 85 63 e8 d0 da c5 62   71
cf99 : 68 e5 63 50 02 49 80 10   63
cfa1 : 13 8a a8 86 ad b1 fb 48   dd
cfa9 : 88 b1 fb c8 91 fb 68 88   cd
cfb1 : 91 fb e6 ae e4 ac d0 b6   34
cfb9 : a5 ae d0 a8 a6 8c e8 ca   06
cfc1 : 8a a8 b1 fb 48 8a 0a 09   9e
cfc9 : 01 90 04 e6 61 e6 fc a8   83
cfd1 : 68 91 fb a9 00 88 91 fb   b8
cfd9 : a5 61 f0 04 c6 61 c6 fc   78
cfe1 : 8a d0 dc 60 a5 01 29 fe   1b
cfe9 : 85 01 a0 07 b1 fd 31 fb   c0
cff1 : 91 fb 88 10 f7 a5 01 09   67
cff9 : 01 85 01 60 00 ff 00 00   09
```

Listing 9. MSE-Listing der Assembler-Routinen aus Listing 6 und 7. Bitte mit dem MSE (Seite 8) eingeben.

```
10 REM        SHAPES DEMO                              <146>
20 :                                                   <252>
30 REM      RICHARD L. RYLANDER       11/23/84
   (REVISED 1/20/85 TO ADD LABELING)                   <130>
40 :                                                   <016>
50 GR=49378     :REM GRAPHIK MODUS                     <165>
60 TX=49411     :REM TEXT MODUS                        <103>
70 :                                                   <046>
80 LB=893       :REM LINKE GRENZE                      <252>
90 RB=894       :REM RECHTE GRENZE                     <206>
100 UB=895      :REM OBERE GRENZE                      <090>
110 DB=896      :REM UNTERE GRENZE                     <046>
120 :                                                  <096>
130 REM    FLAGS FUER VERSCHIEDENE GRAFIKMO
   DI                                                  <107>
140 :                                                  <116>
150 SH=838      :REM SCHATTIERUNG: 0=RANDOM
   , 1=HALBTON                                         <252>
160 SC=839      :REM SKALIERUNG:0=NORMAL (1
   :1), 1=SKAL. (4:3) FUER DARSTELLUNG                 <024>
170 LT=898      :REM LICHTEINFALL: 0=NORMAL
   , 1=VON HINTEN BELEUCHTET                           <033>
180 :                                                  <156>
190 BO=53280    :REM  RAHMENFARBE                      <186>
200 :                                                  <176>
210 REM   FUNKTIONSADRESSEN                            <104>
220 :                                                  <196>
230 CL=51979    :REM BITMAP BEREICH LOESCHE
   N                                                   <221>
240 CO=52001    :REM COLORBEREICH FUELLEN              <013>
250 :                                                  <226>
260 SP=52119    :REM  SPHERE                           <160>
270 TR=52141    :REM  RING VON OBEN                    <108>
280 VC=52150    :REM  ZYLINDER (VERTIKALE A
   CHSE)                                               <011>
290 HC=52153    :REM  ZYLINDER (HORIZONTALE
   ACHSE)                                              <167>
```

Listing 10. Demo-Programm »Shapes« ►

```
300 VT=52186    :REM  RINGSCHNITT (VERTIKAL
    E ACHSE)                                <216>
310 HT=52189    :REM  RINGSCHNITT (AXIS HOR
    IZONTALE ACHSE)                         <199>
320 VS=52203    :REM  SPULE (VERTIKALE ACHS
    E)                                      <241>
330 HS=52206    :REM  SPULE (HORIZONTALE AC
    HSE)                                    <064>
340 :                                       <062>
350 REM  EINZELNE SHAPES ZEICHNEN           <124>
360 :                                       <082>
370 POKE SH,1 :REM HALBTONSCHATTIERUNG      <066>
380 POKE SC,1 :REM SKALIERUNG               <121>
390 POKE LT,0 :REM NORMALBELEUCHTUNG        <217>
400 SYS(CL)   :REM BILDSCHIRM LOESCHEN      <077>
410 SYS(CO),17*11+1  :REM FARBKOMBINATION
    - DUNKELGRAU (11) WEISSE PUNKTE (1)     <245>
414 REM  AUF DEN MEISTEN FARBMONITOREN      <150>
420 POKE BO,1 :REM WEISSER RAHMEN           <041>
430 SYS(GR)   :REM GRAFIK EINSCHALTEN       <023>
432 RW=12:CM=14:MD=1:A$="SHAPES DEMO":GOSU
    B 1900:REM  TITEL                       <207>
434 X1=110:Y1=120:X2=210:Y2=120:BC=1:DC=7:
    GOSUB 1700:REM  COLOR TITEL             <190>
440 POKE LB,38:POKE RB,38:POKE UB,38:POKE
    DB,38        :REM SHAPE FENSTER         <107>
450 SYS(SP),40,199,38                       <129>
455 RW=9:CM=2:MD=1:A$="KUGEL":GOSUB 1900    <102>
460 SYS(HC),120,199,38,38                   <152>
465 CM=11:A$="H-ZYLND.":GOSUB 1900          <250>
470 SYS(VC),200,199 :REM VOREINGESTELLTE P
    ARAMETER WERDEN BENUTZT                 <229>
475 CM=21:A$="V-ZYLIND.":GOSUB 1900         <117>
480 SYS(TR),280,199,15,38                   <084>
485 CM=32:A$="{2SPACE}RING":GOSUB 1900      <122>
490 SYS(VT),40,64                           <046>
495 RW=23:CM=1:A$="H-RING":GOSUB 1900       <205>
500 SYS(HT),120,64                          <190>
505 CM=11:A$="V-RING":GOSUB 1900            <155>
510 SYS(HS),200,64,5,100                    <025>
515 CM=22:A$="H-SPULE":GOSUB 1900           <000>
520 SYS(VS),280,64                          <162>
525 CM=32:A$="V-SPULE":GOSUB 1900           <011>
530 POKE 198,0:WAIT 198,1:POKE 198,0        <125>
540 REM  AUF TASTE WARTEN                   <103>
550 :                                       <018>
560 REM  ZWEI "POKALE", EINER MIT HALBTON-
    , DER ANDERE MIT "RANDOM"-SCHATTIERUNG  <170>
570 :                                       <038>
580 SYS(CL):SYS(CO),16*11+1                 <084>
582 RW=14:CM=14:A$="VERGLEICH":GOSUB 1900:
    RW=15:CM=16:A$="ZWEIER":GOSUB 1900      <228>
584 RW=16:CM=12:A$="SCHATTIERUNGEN":GOSUB
    1900:RW=18:CM=14:A$="<-- HALBTON"       <162>
586 GOSUB 1900:RW=20:CM=15:A$="RANDOM -->"
    :GOSUB 1900                             <217>
590 POKE LB,255:POKE RB,255:POKE UB,49:POK
    E DB,255: REM AN DEN OBEREN RAND        <229>
600 SYS(SP),80,190,80                       <119>
610 POKE UB,51:POKE DB,51: REM WEITERN SPR
    ITE-TEIL ANHAENGEN                      <217>
620 SYS(VS),80,69,10,130                    <170>
630 POKE DB,9:POKE UB,8                     <060>
640 SYS(VT),80,9,25,45                      <168>
650 POKE SH,0  :REM AUF RANDOM SCHATTIERUN
    G UMSCHALTEN                            <228>
660 POKE LB,255:POKE RB,255:POKE UB,49:POK
    E DB,255                                <013>
670 SYS(SP),240,190,80                      <171>
680 POKE UB,51:POKE DB,51                   <130>
690 SYS(VS),240,69,10,130                   <112>
700 POKE DB,9:POKE UB,8                     <130>
710 SYS(VT),240,9,25,45                     <053>
720 POKE 198,0:WAIT 198,1:POKE 198,0        <059>
740 :                                       <208>
750 REM  ZEICHNE "WEIN" SZENE               <144>
760 POKE LT,1 :REM HINTERGRUNGBELEUCHTUNG   <070>
770 POKE SH,1 :REM HALBTONSCHATTIERUNG FUE
    R FLASCHENETIKETT                       <239>
780 SYS(CO):SYS(CL),255: REM BITMAP FUELLE
    N                                       <021>
790 POKE BO,0 : REM SCHWARZER HINTERGRUND   <066>
792 RW=0:CM=0:MD=2:A$="ZEICHENEN MIT":GOSU
    B 1900:RW=1:CM=0:A$="RUECKBELEUCHTUNG"  <050>
794 GOSUB 1900:RW=2:CM=0:A$="GEGEN GESETZT
    EN":GOSUB 1900                          <034>
796 RW=3:CM=0:A$="HINTERGRUND":GOSUB 1900   <169>
798 RW=1:CM=26:A$="FARBEN WERDEN":GOSUB 19
    00                                      <210>
800 RW=2:CM=26:A$="NACHGEZEICHNET":GOSUB 1
    900                                     <060>
810 REM  ZEICHNE FLASCHE                    <187>
820 POKE UB,0:POKE DB,255:POKE LB,255:POKE
    RB,255                                  <087>
830 SYS(VT),150,10,30,50                    <164>
840 POKE UB,255:SYS(VC),150,70,50,60        <113>
850 POKE DB,0:SYS(VT),150,130,6,50          <053>
860 POKE DB,55:POKE UB,0:SYS(VS),150,204,1
    5,181                                   <063>
870 POKE UB,255:SYS(VC),150,221,16,17       <164>
880 :                                       <094>
890 REM  ZEICHENE WEINGLAS                  <127>
900 POKE UB,20:SYS(SP),80,120,60            <235>
910 POKE UB,35:POKE DB,34:SYS(VS),80,34,10
    ,110                                    <183>
920 :                                       <134>
930 REM  ZEICHENT EIN PAAR TRAUBEN          <156>
940 SYS(SP),8,8,8:                          <167>
950 SYS(SP),20,8:SYS(SP),40,8:SYS(SP),12,2
    0:SYS(SP),30,20:SYS(SP),25,16           <139>
960 :                                       <174>
970 REM ZEICHNET APFEL BESTEHEND AUS 2 RIN
    GEN UND EINEM KUGELFRAGMENT             <133>
980 POKE UB,255:POKE DB,255:POKE LB,255:PO
    KE RB,59                                <012>
990 SYS(VT),260,29,0,50:SYS(VT),260,79      <217>
1000 POKE UB,43:POKE DB,43:SYS(SP),260,54,
     60                                     <014>
1010 REM   APFELSTIEL ZEICHNEN              <236>
1020 POKE RB,0:POKE DB,0:SYS(TR),272,104,1
     0,15                                   <173>
1030 REM   BLATT ZEICHNEN                   <113>
1040 POKE DB,255:POKE RB,0:SYS(SP),256,119
     ,15                                    <165>
1050 REM  "RANDOM"-SCHATTIERTES ETTIKETT H
     INZUFUEGEN !                           <191>
1060 POKE UB,255:POKE RB,255:POKE LB,6      <010>
1070 POKE SH,0:SYS(VC),150,72,50,48         <225>
1080 :                                      <040>
1090 REM  BILD NACHCOLORIEREN               <148>
1100 SYS(CO),12                             <033>
1110 X1=200:Y1=1:X2=315:Y2=100:DC=0:BC=2:G
     OSUB 1700                              <145>
1120 X1=240:Y1=110:X2=255:Y2=150:BC=5:GOSU
     B 1700                                 <229>
1130 X1=260:Y1=110:X2=270:Y2=135:BC=9:GOSU
     B 1700                                 <055>
1140 X1=1:Y1=1:X2=48:Y2=30:BC=4:GOSUB 1700  <007>
1150 X1=140:Y1=205:X2=180:Y2=235:BC=7:GOSU
     B 1700                                 <228>
1160 X1=145:Y1=25:X2=195:Y2=115:BC=6:GOSUB
     1700                                   <234>
1170 POKE 198,0:WAIT 198,1:POKE 198,0       <001>
1190 :                                      <150>
1200 REM    "KAFFEE UND PLAETZCHEN"         <110>
1210 POKE SH,0 :REM RANDOM-SCHATTIERUNG FU
     ER PLAETZCHEN                          <230>
1220 SYS(CO),16*11+1:SYS(CL):POKE BO,1      <084>
1230 POKE LB,255:POKE RB,255:POKE UB,255:P
     OKE DB,255 :REM KEIN SHAPE-FENSTER     <040>
1240 SYS(VT),60,20,20,60                    <069>
1250 POKE RB,29:SYS(VT),99,60:POKE RB,255   <198>
1260 SYS(TR),188,180                        <166>
1270 REM HALBTONSCHATTIERTE KAFFEETASSE HI
     NZUFUEGEN.                             <239>
1280 POKE SH,1:POKE UB,0:SYS(VT),188,20:PO
     KE UB,255                              <251>
1290 POKE DB,0:SYS(TR),278,110,20,40        <210>
1300 POKE DB,255:POKE UB,0:POKE LB,0        <140>
1310 SYS(TR),248,90,50,70:POKE LB,255:SYS(
     SP),248,110,10:POKE UB,255             <160>
1320 SYS(VC),308,100,10,10                  <246>
1330 SYS(VC),188,77,60,57                   <154>
1340 POKE DB,0:SYS(VT),188,134,40,60        <050>
1360 SYS(CO),1+16*9 :REM 1=WEISSER HINTERG
     RUND, 9=BRAUNE PUNKTE                  <067>
1370 X1=130:Y1=1:X2=319:Y2=136:BC=1:DC=5:G
     OSUB 1700                              <009>
1380 X1=250:Y1=144:X2=319:Y2=144:GOSUB 170
     0                                      <161>
1390 POKE 198,0:WAIT 198,1:POKE 198,0       <223>
1410 :                                      <116>
1420 REM  ZEICHNE "VERBUNDENE" RINGE DURCH
     UEBERLAPPUNG DER FENSTER               <138>
1430 POKE LT,0 :REM BLAUE PUNKTE AUF WEISS
     , KEINE HINTERGRUNDBELEUTUNG           <041>
1440 SYS(CL):SYS(CO),1+16*6: REM 1=WEISSER
```

Listing 10. Demo-Programm »Shapes« (Fortsetzung)

```
        INTERGRUND, 6=BLAUE PUNKTE                <255>
1450 POKE UB,255:POKE DB,255:POKE LB,255:P
     OKE RB,255: REM KEINE FENSTER                <058>
1460 POKE SH,0  :REM ZUFAELLIGE SCHATTIERU
     NG                                           <096>
1470 SYS(TR),244,84,48,70                         <088>
1480 SYS(TR),160,84:SYS(TR),76,84                 <109>
1490 SYS(TR),118,156:SYS(TR),202,156              <163>
1500 REM  UEBERLAPPENDE ZONEN HINZUFUEGEN         <129>
1510 POKE RB,0:POKE DB,0:SYS(TR),160,84:PO
     KE RB,255:POKE LB,0                          <127>
1520 SYS(TR),76,84:POKE DB,255:POKE UB,0:S
     YS(TR),118,156                               <198>
1530 POKE LB,255:POKE RB,0:SYS(TR),202,156
     :POKE LB,27                                  <081>
1540 POKE DB,0:POKE UB,255:SYS(TR),160,84         <152>
1550 POKE DB,0:POKE UB,255:SYS(TR),160,84         <162>
1560 POKE LB,255:POKE UB,27:SYS(TR),244,84        <171>
1570 POKE LB,0:POKE RB,27:POKE UB,255:SYS(
     TR),244,84                                   <180>
1580 POKE 198,0                                   <220>
1590 GET A$:IF A$= "" THEN 1590                   <184>
1600 SYS(TX):POKE BO,14:REM RUECKKEHR ZUM
     TEXTMODUS                                    <239>
1610 END                                          <088>
1620 :                                            <072>
1630 REM UNTERPROGRAMM ZU NACHCOLORIEREN V
     ERSCHIEDENER ZONEN                           <233>
1640 REM ACHTUNG: ES KANN JEWEILS NUR EINE
      RECHTECKIGE FLAECHE                         <232>
1650 REM VON 8X8 PUNKTEN EINGEFAERBT WERDE
     N !!                                         <088>
1660 REM (X1,Y1)=UNTERER LINKER PUNKT, (X2
     ,Y2)=OBERER RECHTER PUNKT                    <235>
1690 REM DIE Y-KOORDINATEN MUESSEN "UNSKAL
     LIERT" SEIN. FALLS SCALE-FLAG GESETZT
     :                                            <223>
1700 IF PEEK(SC)THEN Y1=(Y1+1)*213/256:Y2=
     (Y2+1)*213/256                               <168>
1710 REM UEBERGABEVARIABLE IST CC.                <236>
1720 REM CC=16*DC + BC [DC=PUNKT FARBE, BC
     = HINTERGRUNDFARBE]                          <037>
1730 CC=16*DC+BC                                  <227>
1740 FOR IX=INT(X1/8) TO INT(X2/8)                <033>
1750 FOR IY=INT(Y1/8) TO INT(Y2/8)                <061>
1760 POKE 34752+IX-40*IY,CC                       <092>
1770 NEXT:NEXT:RETURN                             <227>
1780 :                                            <232>
1790 REM "TEXT"-UNTERROUTINE ERMOEGLICHT E
     INFUEGEN VON TEXT IN DIE BILDER.             <076>
1800 REM "RW" UND "CM" SIND DIE ZEILEN(0-2
     4) UND SPALTEN(0-39) KOORDINATEN DES         <040>
1810 REM ERSTEN BUCHSTABEN DES ZU DRUCKEND
     EN TEXTSTRINGS.                              <249>
1820 REM UEBERGABEVARIABLE FUER TEXT IST A
     $                                            <073>
1830 REM "MD" ZEIGT DEN DRUCKMODUS AN:            <133>
1840 REM  1 - NORMAL ("SCHWARZE" BUCHSTABE
     N AUF "WEISSEM" HINTERGRUND)                 <206>
1850 REM  2 - INVERSE ("WEISSE" BUCHSTABEN
      AUF "SCHWARZEM" HINTERGRUND)                <103>
1860 REM  3 - BUCHSTABEN WERDEN MIT LOGISC
     HEM ODER MIT HINTERGRUND VERKNUEPFT.         <191>
1870 REM  4 - BUCHSTABEN WERDEN MIT LOGISC
     HEM UND MIT HINTERGRUND VERKNUEPFT.          <070>
1880 REM  5 - BUCHSTABEN WERDEN MIT LOGISC
     HEM EXOR MIT HINTERGRUND VERKNUEPFT.         <111>
1890 :                                            <088>
1900 SB=40952:TB=54272:IF(MD AND 1)THEN TB
     =53248:REM BASISADRESSEN                     <044>
1910 OS=320*RW+8*CM:REM  ADRESSE DES ZEICH
     ENGENERATORS                                 <111>
1920 POKE 56334,PEEK(56334)AND 254:REM DIS
     ABLE IRQ TIMER                               <136>
1930 POKE 1,PEEK(1)AND 251:REM CHARACTER R
     OM EINSCHALTEN                               <066>
1940 L=LEN(A$):FOR N=1 TO L:N8=N*8+OS+SB          <068>
1950 X=ASC(MID$(A$,N,1)):IF X>63 THEN X=X-
     64                                           <107>
1960 TC=TB+8*X                                    <173>
1970 ON MD GOTO 1980,1980,1990,2000,2010          <120>
1980 POKE 53231,36:GOTO 2020                      <028>
1990 POKE 53231,17:GOTO 2020                      <038>
2000 POKE 53231,49:GOTO 2020                      <062>
2010 POKE 53231,81                                <044>
2020 POKE 252,N8/256:POKE 251,N8-256*INT(N
     8/256)                                       <137>
2030 POKE 254,TC/256:POKE 253,TC-256*INT(T
     C/256)                                       <136>
2040 SYS(53221):NEXT                              <158>
2050 POKE 1,PEEK(1)OR 4:POKE 56334,PEEK(56
     334)OR 1:REM NORMAL SCHALTEN                 <136>
2060 RETURN                                       <086>
```

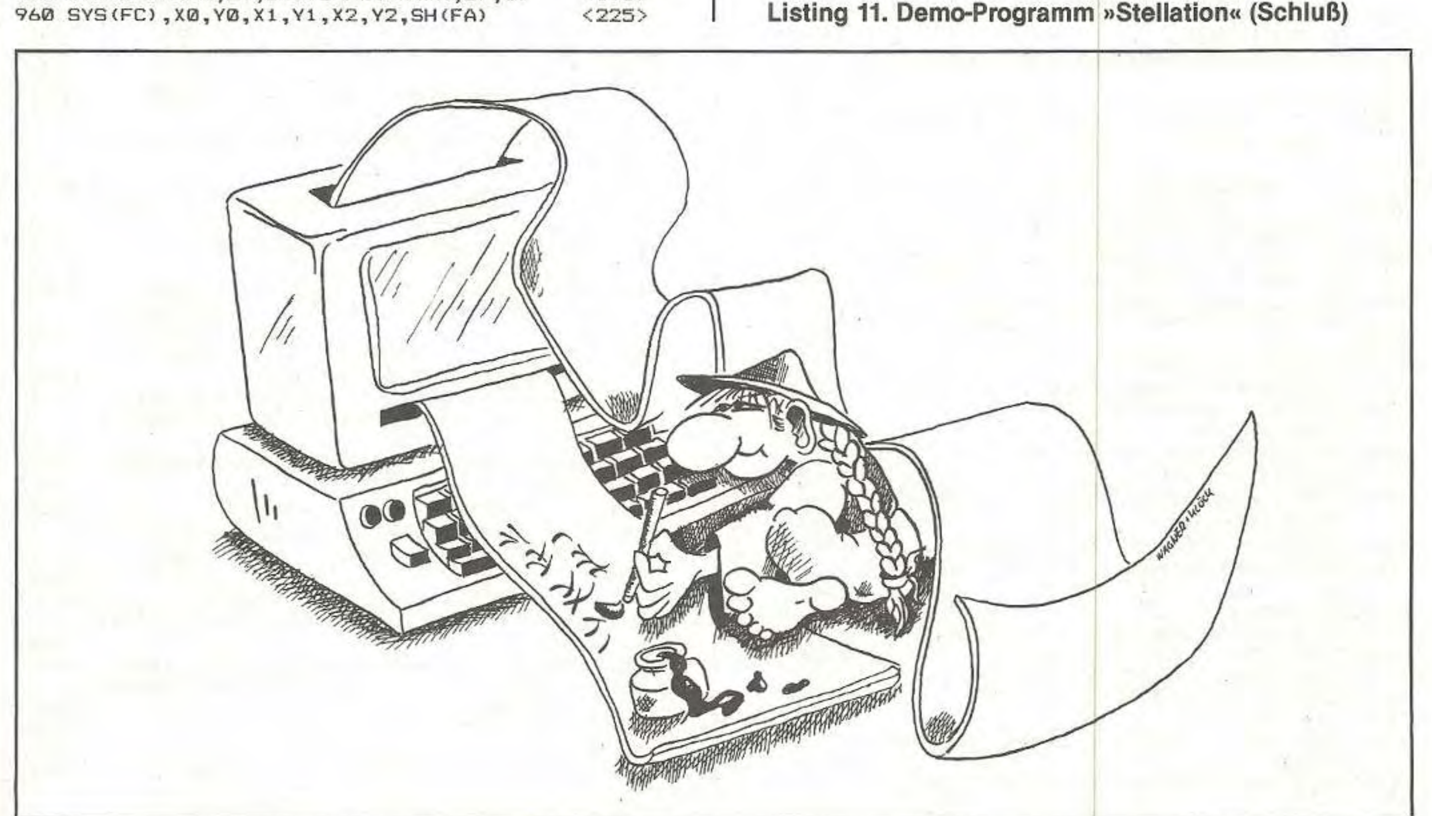

Listing 10. Demo-Programm »Shapes« (Schluß)

```
10 REM "STELLATION"                              <061>
20 REM  BY RICHARD L. RYLANDER       12/5/84     <185>
30 :                                             <006>
40 GR=49378     :REM GRAPHIK MODUS               <155>
50 TX=49411     :REM TEXT MODUS                  <093>
60 BO=53280     :REM RAHMENFARBE                 <056>
70 :                                             <046>
80 REM  STYLE PARAMETERS                         <157>
90 POKE 839,1  :REM SKALIERUNG (3:4)             <047>
100 POKE 871,0 :REM FACETTENRAENDER (0=NAC
    HZEICHNEN, 1=LOESCHEN)                       <086>
110 SH=838      :REM SCHATTIERUNG (0=RANDOM
    , 1=HALBTON)                                 <162>
120 EG=868      :REM RANDFLAG (0=NORMAL, 1=
    LINIEN AM RAND HINZUFUEGEN)                  <058>
130 :                                            <106>
140 REM   FUNKTIONS ADRESSEN                     <034>
150 CL=51979    :REM BITMAP LOESCHEN             <242>
160 CO=52001    :REM COLOR MAP FUELLEN           <112>
170 FC=52052    :REM SCHATTIERTE FACETTEN Z
    EICHNEN                                      <124>
180 KS=53081    :REM SORTIERROUTINE              <231>
190 :                                            <166>
200 XC=160:YC=120  :REM ZENTRIERUNGSKOORDI
    NATEN                                        <164>
210 :                                            <186>
220 PRINT"{CLR}****************************
    ************"                                <253>
230 PRINT"* KLEINER STERNFOERMIGER DODEKAH
    EDRON *"                                     <133>
240 PRINT"*********************************
    *******"                                     <024>
250 PRINT"{2DOWN,SPACE}SCHATTIERUNG:"            <037>
260 PRINT" R=RANDOM, H=HALBTON"                  <001>
270 INPUT"{DOWN,SPACE}IHRE WAHL{5SPACE}H{3
    LEFT}";A$                                    <001>
280 POKE SH,0:IF A$="H" THEN POKE SH,1           <206>
290 PRINT"{2DOWN,SPACE}RAND-STIL :"              <230>
300 PRINT"{DOWN,SPACE}N - NORMAL":PRINT"{D
    OWN,SPACE}E - ECKENBETONUNG"                 <196>
310 PRINT"{DOWN,SPACE}W - DRAHTGITTER"           <251>
320 INPUT"{DOWN,SPACE}IHRE WAHL{5SPACE}N{3
    LEFT}";A$                                    <243>
330 POKE EG,0:WI=0:IF A$="N" THEN 360            <101>
340 POKE EG,1:IF A$="W" THEN WI=-1               <252>
350 :                                            <072>
360 PRINT"{DOWN,SPACE}LESE SCHEITEL DATEN"       <003>
370 VN=32:DIM P%(VN-1,2)                         <088>
380 FOR N=0 TO VN-1:READ P%(N,0),P%(N,1),P
    %(N,2):NEXT                                  <242>
390 :                                            <112>
400 PRINT"{DOWN,SPACE}GEBEN SIE X, Y, UND
    Z WINKEL EIN"                                <080>
410 PRINT"{7SPACE}(WINKEL IN GRAD)"              <102>
420 INPUT X,Y,Z                                  <163>
430 J=3.14159265/180:X=X*J:Y=Y*J:Z=Z*J           <104>
440 X0=COS(Y)*COS(Z)-SIN(X)*SIN(Y)*SIN(Z):
    X1=COS(Y)*SIN(Z)+SIN(X)*SIN(Y)*COS(Z)        <148>
450 X2=-COS(X)*SIN(Y):Y0=-COS(X)*SIN(Z):Y1
    =COS(X)*COS(A):Y2=SIN(X)                     <074>
460 Z0=SIN(Y)*COS(Z)+SIN(X)*COS(Y)*SIN(Z)        <138>
470 Z1=SIN(Y)*SIN(Z)-SIN(X)*COS(Y)*COS(Z):
    Z2=COS(X)*COS(Y)                             <250>
480 PRINT"{DOWN,SPACE}ROTATIONSBERECHNUNG"       <202>
490 FOR N=0 TO VN-1                              <082>
500 X=P%(N,0):Y=P%(N,1):Z=P%(N,2)                <027>
510 P%(N,0)=X0*X+X1*Y+X2*Z:P%(N,1)=Y0*X+Y1
    *Y+Y2*Z:P%(N,2)=Z0*X+Z1*Y+Z2*Z:NEXT          <136>
520 :                                            <244>
530 FA=60: REM  ANZAHL DER FACETTEN              <201>
540 DIM F%(FA/2,2),SH(FA/2),Z%(FA/2),K%(FA
    /2)                                          <099>
550 PRINT" LESE VERBINDUNGSDATEN{14SPACE}"       <089>
560 VF=-1: REM  VF = ANZAHL DER SICHTBAREN
     FACETTEN                                    <057>
570 FOR N=1 TO FA                                <148>
580 VF=VF+1                                      <223>
590 FOR I=0 TO 2 :READ F%(VF,I):NEXT             <042>
```

Listing 11. Demo-Programm »Stellation«

```
600 REM  BERECHNUNG DER NORMALENVEKTOREN     <118>
610 Z=(P%(F%(VF,2),0)-P%(F%(VF,1),0))*(P%(
    F%(VF,0),1)-P%(F%(VF,1),1))              <150>
620 Z=Z-(P%(F%(VF,0),0)-P%(F%(VF,1),0))*(P
    %(F%(VF,2),1)-P%(F%(VF,1),1))            <141>
630 IF Z=<0 THEN 720:REM  FACETTE NICHT SI
    CHTBAR                                   <009>
640 X=(P%(F%(VF,2),1)-P%(F%(VF,1),1))*(P%(
    F%(VF,0),2)-P%(F%(VF,1),2))              <178>
650 X=X-(P%(F%(VF,0),1)-P%(F%(VF,1),1))*(P
    %(F%(VF,2),2)-P%(F%(VF,1),2))            <155>
660 Y=(P%(F%(VF,2),2)-P%(F%(VF,1),2))*(P%(
    F%(VF,0),0)-P%(F%(VF,1),0))              <199>
670 Y=Y-(P%(F%(VF,0),2)-P%(F%(VF,1),2))*(P
    %(F%(VF,2),0)-P%(F%(VF,1),0))            <183>
680 NC=SQR(X*X+Y*Y+Z*Z):REM  LAENGE DES NO
    RMALENVEKTORS                            <068>
690 SH(VF)=26*(2*Z+X+Y)/NC                   <114>
700 SH(VF)=(SH(VF)+64)*(SH(VF)+64)/256:REM
     SCHATTIERUNG                            <177>
710 GOTO 730                                 <226>
720 VF=VF-1                                  <235>
730 NEXT                                     <232>
740 :                                        <208>
750 PRINT" BERECHNUNG DER DARSTELLUNGSGROE
    SSE"                                     <140>
760 Y=0:FOR N=0 TO VN-1:IF ABS(P%(N,1))>Y
    THEN Y=ABS(P%(N,1))                      <190>
770 NEXT:S=119/Y                             <075>
780 FOR N=0 TO VN-1:P%(N,1)=S*P%(N,1)+YC:P
    %(N,0)=S*P%(N,0)+XC:NEXT                 <130>
790 :                                        <004>
800 :                                        <119>
810 FOR N=0 TO VF                            <115>
820 Z%(N)=(P%(F%(N,0),2)+P%(F%(N,1),2)+P%(
    F%(N,2),2))/3:NEXT                       <060>
830 :                                        <044>
840 PRINT" SORTIERUNG DER FACETTEN"          <010>
850 POKE 140,VF                              <004>
860 K%(0)=K%(0):POKE 251,PEEK(71):POKE 252
    ,PEEK(72)                                <154>
870 Z%(0)=Z%(0):POKE 253,PEEK(71):POKE 254
    ,PEEK(72)                                <019>
880 SYS(KS)                                  <153>
890 :                                        <104>
900 REM ZEICHNE FACETTEN                     <002>
910 SYS(GR):SYS(CO):SYS(CL):POKE BO,1        <168>
920 FOR N=0 TO VF:FA=K%(N)                   <009>
930 IF WI THEN SH(FA)=64                     <016>
940 X0=P%(F%(FA,0),0):Y0=P%(F%(FA,0),1):X1
    =P%(F%(FA,1),0):Y1=P%(F%(FA,1),1)        <170>
950 X2=P%(F%(FA,2),0):Y2=P%(F%(FA,2),1)      <193>
960 SYS(FC),X0,Y0,X1,Y1,X2,Y2,SH(FA)         <225>
970 NEXT                                     <218>
980 POKE 198,0                               <126>
990 GET A$:IF A$="" THEN 990                 <168>
1000 SYS(TX):POKE BO,14:END                  <128>
1010 :                                       <224>
1020 REM    SCHEITELDATEN                    <083>
1030 DATA 1000,618,0, 1000,-618,0, -1000,6
     18,0, -1000,-618,0                      <092>
1040 DATA 0,1000,618, 0,1000,-618, 0,-1000
     ,618, 0,-1000,-618                      <099>
1050 DATA 618,0,1000, -618,0,1000, 618,0,-
     1000, -618,0,-1000                      <183>
1060 DATA 618,0,236, 618,0,-236, -618,0,23
     6, -618,0,-236                          <043>
1070 DATA 236,618,0, -236,618,0, 236,-618,
     0, -236,-618,0                          <243>
1080 DATA 0,236,618, 0,-236,618, 0,236,-61
     8, 0,-236,-618                          <100>
1090 DATA 382,382,382, 382,382,-382, 382,-
     382,382, 382,-382,-382                  <236>
1100 DATA -382,382,382, -382,382,-382, -38
     2,-382,382, -382,-382,-382              <056>
1110 :                                       <070>
1120 REM    VERBINDUNGSDATEN                 <109>
1130 DATA 0,12,13, 0,13,25, 0,25,16, 0,16,
     24, 0,24,12                             <057>
1140 DATA 1,12,26, 1,26,18, 1,18,27, 1,27,
     13, 1,13,12                             <248>
1150 DATA 2,15,14, 2,14,28, 2,28,17, 2,17,
     29, 2,29,15                             <251>
1160 DATA 3,14,15, 3,15,31, 3,31,19, 3,19,
     30, 3,30,14                             <208>
1170 DATA 4,16,17, 4,17,28, 4,28,20, 4,20,
     24, 4,24,16                             <110>
1180 DATA 5,17,16, 5,16,25, 5,25,22, 5,22,
     29, 5,29,17                             <168>
1190 DATA 6,19,18, 6,18,26, 6,26,21, 6,21,
     30, 6,30,19                             <177>
1200 DATA 7,18,19, 7,19,31, 7,31,23, 7,23,
     27, 7,27,18                             <235>
1210 DATA 8,20,21, 8,21,26, 8,26,12, 8,12,
     24, 8,24,20                             <060>
1220 DATA 9,21,20, 9,20,28, 9,28,14, 9,14,
     30, 9,30,21                             <144>
1230 DATA 10,23,22, 10,22,25, 10,25,13, 10
     ,13,27, 10,27,23                        <074>
1240 DATA 11,22,23, 11,23,31, 11,31,15, 11
     ,15,29, 11,29,22                        <224>
```


Listing 11. Demo-Programm »Stellation« (Schluß)



Bestellungen aus anderen Ländern bitte per Auslandspost-anweisung! Achtung: Nicht die eingeheftete Zahlkarte verwenden.

Bestellungen aus der Schweiz bitte direkt an: Markt&Technik Vertriebs AG, Kollerstr. 3, CH-6300 Zug, Tel.: 042/223155.

Bestellungen aus Österreich bitte direkt an: Bücherzentrum Meidling, Schönbrunnerstr. 261, 1120 Wien.
Microcomput-ique Erhard Schiller Fasangasse 21 1030 Wien

Programme aus früheren Ausgaben

Ausgabe 8/85
Bestell-Nr. L 6 8508 A DM 29,90*

Sonderheft: Tips & Tricks (2. ü. Auflage)
Floppy-Utilities
CB 023 DM 29,90*
Hilfsprogramme
CB 024 DM 29,90*

Sonderheft: Spiele
Beide Disketten in einem Paket!
Verwenden Sie **nur** diese Bestell-Nr.:
L 6 85S3 A **DM 34,90***

Ausgabe 7/85
Bestell-Nr. L 6 8507 A **DM 29,90***
Commodore 64
Haushaltsbuch (AdM) S. 57
Terminalprogramm S. 152
Centron S. 80
Editor S. 151
Ein-/Ausgaberoutine S 77
Fenster (C 16) S 84
File-Compactor S. 82
Hypra-Assembler (LdM) S. 66
IEEE-Basic S. 46
Logik S. 144
Merkzettel S. 83
Modulator S. 46
REM-Killer S. 75
Sound Editor S. 136
Startgenerator S. 74

Ausgabe 6/85
Bestell-Nr. L 6 8506A **DM 29,90***
Commodore 64
MSE S. 54
HI-EDDI/MPS 801 S. 69
Prost S. 76
E-Routine 64 S. 148
GCR-HEX S. 117
HEX-GCR S. 118
Samurai S. 72
Scroll-Machine (LdM) S. 61
Crossreferenz S. 155
Heapsort S. 126

C 16
F-Plotter S. 68

Sonderheft: Abenteuerspiele
Bestell-Nr.: L685 S2 **DM 34,90***

Ausgabe 5/85
Bestell-Nr. L 6 8505 A **DM 29,90***
Commodore 64
Checksum. Schnell S. 54
MSE Lader S. 55
MPS 802 S. 31
Format-System S. 147
VIC S. 175
6510 I S. 71
Sternenhimmel (AdM) S. 57
Assemblerkurs S. 144
Direktory-Sorter S. 77
Trick.OBJ S. 65
3D-Movie-Maker (LdM) S. 65
Modulator (Heft 4) S. 155
VC 20
Checksummer S. 54
Minigrafik S. 69
Longscreen S. 83
C 16
Help & Trace S. 84

Ausgabe 4/85
Bestell-Nr. L 6 8504A **DM 29,90***
Commodore 64
Checksum. Schnell S. 54
MSE Lader S. 58
S-MON Teil 5 S. 67
Window 64 S. 88
XBASIC S. 59
ASS. Kurs Teil 8 S. 138
Hypra-Perfect S. 76
11 neue Einzeiler S. 153
Print-List S. 79
Netzgrafik S. 71
ROT-DEMO S. 73
X.Y-DEMO 1 u. 2 S. 73
Ostereier S. 51
Befehlserweiterung S. 23
VC 20
Checksummer S. 56
Supergrafik III S. 73
Prg. Schutz S. 83

Ausgabe 3/85
Bestell-Nr. L 6 8503A **DM 29,90***
Commodore 64
Checksu. Schnell S. 70
MSE Lader S. 79
Hires-3/Teil 2 S. 136
Intellisearch$C S. 153
Son of Destroyer S. 42
Ligatab (LdM) S. 63
Fl. Basicloader S. 82
Floppylister S. 82
E.Zeichnen 1 und 2 S. 72
Meisterschütze (AdM) S. 53
Saver S. 89
Screensave S. 88
Pic-Loader S. 90
VC 20
Checksummer S. 70
IRQ-Clock S. 156
Autostart S. 159

Ausgabe 2/85
Bestell-Nr. L 6 8502A **DM 29,90***
Commodore 64
Checksummer 64 S. 65
MSE S. 68
SMON (Teil 4) S. 72
Grab des Pharao (LdM) S. 56
Basic-Lader S. 91
RAM-Floppy S. 92
Notlandung S. 156
Hires-3 S. 123
Benchmark-Test S. 37
Format.Stringkur S. 148
Versch. Test.USR S. 140
Cursorsteuerung S. 86
Rolling-Data S. 88
Super-Memory S. 81
VC 20
Checksummer VC 20 S. 65
Familienplanung (AdM) S. 53
Super 8-Steuerung S. 70
Q+Bert (3K) S. 78
Joypaint S. 143
Bitmapping S. 145
Multicolor-Demo S. 146

Ausgabe 1/85
Bestell-Nr. L 6 8501A **DM 29,90***
Commodore 64
Checksummer 64 S. 72
Handballtrainer (AdM) S. 53
SMON (Teil 3) S. 69
Hi-Eddi (LdM) S. 57
Hypra-Load mal vier S. 82
Tips und Tricks S. 87
Provic 64 S. 76
Eingabe (UPB) S. 156
VC 20
Checksummer VC 20 S. 72

Ausgabe 12/84
Bestell-Nr. CB 022 **DM 29,90***
Commodore 64
Synthesizer (AdM) S. 51
SMON (2. Teil) S. 60
3D-Vier gewinnt S. 96
Trace S. 76
Stringy S. 88
Lader S. 92
Auto S. 84
Listschutz S. 85
Simons Axo (SB) S. 64
Kreuzworträtsel S. 150

VC 20
Mathematikal Basic (8K >) (LdM) S. 55
Fast Tape S. 80

Ausgabe 11/84
Bestell-Nr. CB 020 **DM 29,90***

Ausgabe 10/84
Bestell-Nr. CB 019 **DM 29,90***

Ausgabe 9/84
Bestell-Nr. CB 014 **DM 29,90***

Ausgabe 8/84
Bestell-Nr. CB 013 **DM 29,90***

Bedeutung der Abkürzungen

*LdM = Listing des Monats
*AdM = Anwendung des Monats
*SB = Simons Basic
*GV = Grundversion
*GV > = alle Speicherversionen können verwendet werden (einschließlich GV)
*3K = 3-KByte-Speichererweiterung wird benötigt
*8K > = Speichererweiterung größer als 8 KByte wird benötigt
*UPB = Unterprogrammbibliothek

* Alle Preise inklusive Mehrwertsteuer.

Achtung!
Bitte verwenden Sie für Ihre Bestellung **nur** die eingeheftete Postscheck-Zahlkarte zur Überweisung des Rechnungsbetrags.

Fehlende Hefte erhalten Sie bei: Markt & Technik Vertrieb 64'er Hans-Pinsel-Str. 2, 8013 Haar

Soft-Scrolling auf dem C 64

Soft-Scrolling in eigene Programme einzubauen, kann oft zum Problem werden. Wie es gemacht wird, zeigt Ihnen dieser Beitrag.

Jeder kennt bestimmt Defender oder ähnliche Arcade-Games, bei denen die Hintergrundlandschaft über den Bildschirm zieht. Gemeint ist das sogenannte Smooth- oder Soft-Scrolling. Der Unterschied zwischen normalem und Soft-Scrolling ist, daß der Bildschirm beim Soft-Scrolling in Einerschritten (um ein Pixel) und beim normalen in Achterschritten (immer um acht Pixel gleichzeitig) verschoben wird.

Beim Achterscrolling werden einfach die Speicherstellen des Video-RAMs (ab 1024/$0800) um jeweils eine Stelle nach links, rechts, oben oder unten verschoben.

Zum Achterscrolling ein Beispiel in Basic:

```
10  REM Beispiel Nr. 1
20  PRINT CHR$(147);TAB(14);
30  PRINT "ABCDEFGHIJKLMNOPQRSTUVWXYZ"
40  FOR I=1064 TO 1024 STEP -1
50  POKE I,PEEK (I-1)
60  NEXT
70  GOTO 40
```


Dieses Programm arbeitet zwar noch relativ schnell, aber wir werden später diese Aufgabe einem Assemblerprogramm übergeben.

Das Alphabet wird nach dem Eintippen dieses Programms nach rechts herausgeschoben. Dieses Achterscrolling ist aber zu sprunghaft, und falls es schneller läuft, ist die gescrollte Schrift unleserlich.

Das Soft-Scrolling bewirkt, daß die Schrift gleichmäßig fließend und damit leserlich gemacht wird. Wie scrollt man denn nun aber den Bildschirm um nur ein Pixel?

Die Hauptrolle hierbei spielen die Speicherstellen 53248+17 (53248 = Basisadresse des Video-Chips +17 für das vertikale und 53248+22 für das horizontale Scrolling).

Befassen wir uns zunächst mit dem horizontalen Scrolling. Die Speicheradresse 53248+22 ist folgendermaßen aufgebaut:

Bit 7 bis 5 : Sind nicht benutzt
Bit 4 : Zur Umschaltung auf Multicolor
Wichtig sind für unsere Anwendung nur:
Bit 3 : Umschalten von 40 auf 38 Spalten
Bit 0 bis 2 : Horizontales Scrolling

Dazu ein Versuch:

POKE 53248+22,PEEK (53248+22) AND 248 OR 3

Nun müßte sich der Bildschirm um drei Pixels nach rechts verschoben haben.

Einige werden sich jetzt fragen, warum die ANDs und ORs?

Nehmen wir einmal an, im Register 22 des VIC würde die Zahl 107 stehen. Diese Zahl wird mit 248 AND-verknüpft. Beide Zahlen müssen binär aufgeschlüsselt werden:

```
    107 = 01101011
AND 248 = 11111000
-------------------
   ergibt 01101000
```

Durch diese AND-Funktion werden also die Bits 0 bis 2 gelöscht. Die Bits 3 bis 7 ändern sich nie. Durch die OR-Funktion werden die entsprechenden Bits gesetzt:

```
       01101000
OR 3   00000011
---------------
ergibt 01101011
```

Die Bits für das horizontale Scrolling enthalten jetzt den Wert 3. Der höchste mit den Bits 0 bis 2 zu erreichende Wert ist 7, der niedrigste ist 0. Je größer der Wert in diesen drei Bits, desto weiter wird der Bildschirm nach rechts verschoben! Mit dieser Erkenntnis können wir den Bildschirm um sieben Pixels nach rechts rollen!

```
10  REM Beispiel Nr. 2
20  FOR I=0 TO 7
30  POKE 53248+22,PEEK (53248+22) AND 248 OR I
40  FOR T=0 TO 1000:NEXT T
50  NEXT I
60  RUN
```

Dieses Programm verschiebt den Bildschirm langsam von der normalen Position um sieben Pixel nach rechts. Dann wird er wieder auf die normale Position zurückgesetzt und rollt wieder nach rechts. Wie macht man es nun, daß der Bildschirm über das 7. Pixel hinausscrollt?

Dies ist ganz einfach: Nachdem das 7. Pixel erreicht wurde, machen wir ein Achterscrolling, das heißt wir verschieben einen Bildschirmcode um ein Zeichen nach rechts. Gleichzeitig setzen wir das Scrollregister wieder auf Null und haben jetzt das 8. Pixel erreicht.

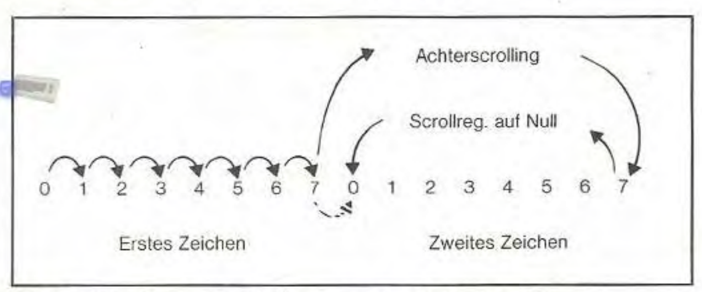


Bild 1. Grafische Erläuterung des Soft-Scrolling

Jetzt können wir die Prozedur wieder von vorne beginnen (dazu ein Flußdiagramm).

S=SCROLLREGISTER

Dieses Flußdiagramm muß nun noch in ein Programm umgesetzt werden. Würde das gesamte Programm in Basic geschrieben, wäre es ziemlich langsam. Deshalb wird im nachstehenden Listing das Achterscrolling in Assembler realisiert. Es scrollt die erste Zeile in Achterschritten nach rechts. Zuerst die Darstellung in Maschinensprache und dann in DATA-Zeilen:

```
  LDA $0427    ; Speicherstelle 1063 merken
  STA $FB
  LDX #38      ; Bei Spalte 38 anfangen
V LDA $0400,X  ; Bildschirmcode laden
  INX          ; Um 1 erhöhen
  STA $0400,X  ; Und abspeichern
  DEX
  DEX          ; Um 2 erhöhen
  CPX #$FF     ; Schon fertig ?
  BNE V
  LDA $FB      ; 1063 wieder nach 1024
  STA $0400
  LDA $D016    ; Scrollregister laden
  AND #248     ; Scrollbits löschen
  STA $D016    ; Zurückspeichern
  RTS
```

Dieses Maschinenprogramm schiebt die Zeilen, die in der ersten Zeile des Bildschirms stehen, nicht nur nach rechts, sondern macht ein Rundum-Scrolling. Die rechts verschwundene Schrift erscheint also wieder am linken Bildschirmrand. Dies wird durch die ersten und letzten beiden Maschinenprogrammzeilen ermöglicht. Startet man dieses Programm, so ist ab und zu ein leichtes Zucken zu erkennen. Dies liegt daran, daß der Elektronenstrahl, der den Bildschirm 25mal in der Sekunde aufbaut, die erste Zeile schreibt, während sie gescrollt wird. Wie kann man das vermeiden?

Es ist das Register 53248+18, das benötigt wird. In ihm steht die Zeile, in der sich der Elektronenstrahl befindet (nähere Informationen siehe 64'er, Ausgabe 11/84).

Wir brauchen also nur dieses Register auszulesen und die erste Zeile erst dann zu scrollen, nachdem sie vollständig vom Elektronenstrahl aufgebaut wurde.

Die Abfrage muß in Assembler geschehen, da ein Basic-Programm im Verhältnis zur Geschwindigkeit des Elektronenstrahls zu langsam ist. Der richtige Zeitpunkt für den Beginn des Scrollings ist ungefähr dann gegeben, wenn der Strahl die Rasterzeile 100 erreicht hat.

Wir müssen das obige Maschinenprogramm also noch etwas erweitern. Ganz am Anfang muß stehen:

```
X LDA  53248+18 ; Rasterzeile auslesen
  CMP  #160     ; Vergleiche auf 160
  BMI  X        ; Wenn kleiner dann nach X
  CMP  #190     ; Vergleiche auf 190
  BPL  X        ; Wenn größer dann nach X
  RTS
```

Und nun das gesamte 1-Zeilen-Soft-Scrolling in Basic: (Das dazugehörige Maschinenprogramm ist so geschrieben, daß es in jeden Bereich gePOKEt werden kann!)

```
10   REM Beispiel Nr. 3
20   PRINT CHR$(147);GOSUB 100
30   PRINT "Beliebiger Text kleiner als 40 Zeichen"
40   REM Farbe = schwarz
50   FOR I=55296 TO 55296+39:POKE I,0:NEXT
60   FOR I=1 TO 7
70   POKE 53248+22,PEEK (53248+22) AND248 ORI
80   FOR T=0 TO 5:NEXT
90   NEXT I
100  FOR T=0 TO 3:NEXT
110  SYS32768:REM Achterscrolling
120  GOTO 60
130  REM Masch.PRG einlesen
140  FOR I=32768 TO 32768+44:REM 44 DATAs
150  READ G:S=S+G:POKE I,G:NEXT
160  IF S=6044 THEN 180
170  PRINT "FEHLER IN DATAS!":STOP
180  RETURN
190  DATA 173,18,208,201,160,48,249,201
200  DATA 190,16,245,173,39,4,141,255,207
210  DATA 162,38,189,0,4,232,157,0,4,202
220  DATA 202,224,255,208,243,173,255,207
230  DATA 141,0,4,173,22,208,41,248,9
240  DATA 0,141,22,208,96
```

Man stellt fest, daß die Zeichen am rechten Rand langsam herauswandern, aber am linken Rand plötzlich auftauchen. Dies läßt sich verhindern, indem man den Bildschirm seitlich verengt (Bit 3 des Registers 53248+22 löschen). Das kann man mit

POKE 53248+22,PEEK(53248+22)AND247 erreichen.

Platz dafür wäre zum Beispiel in Zeile 45.

Basic reicht nicht aus, um eine angemessene Geschwindigkeit zu erreichen. Deshalb steigen wir nun ganz auf Assembler um! Schreiben wir zunächst die Zeilen 60 bis 90 des Beispiels Nummer 3 in Assembler:

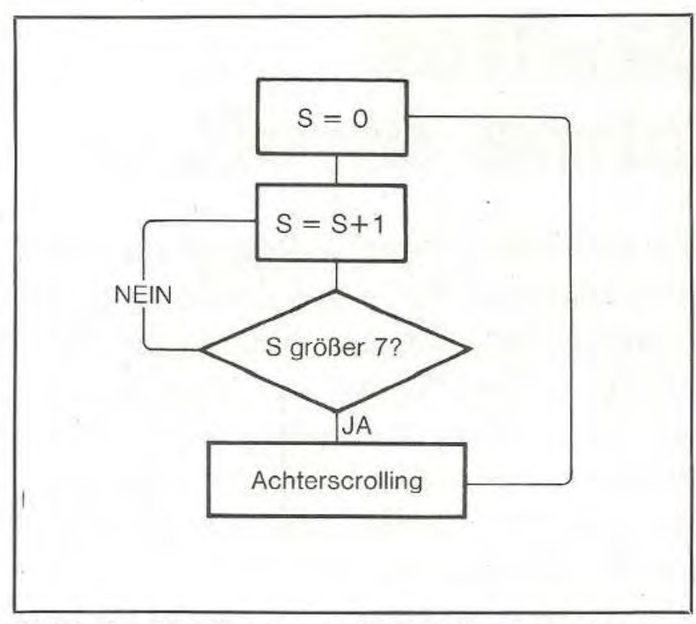


Bild 2. Das Flußdiagramm erläutert die verwendete Scroll-Routine

```
M LDA 53248+22 ; Scrollregister laden
  AND #7       ; Scrollbits ausfiltern
  CMP #7       ; Schon 7 ?
  BNE N        ; Wenn nicht dann Einerscrolling
  JSR ACHTSC   ; Achterscrolling
  JMP M        ; Nächste Verschiebung
N INC 53248+22 ; Scrollreg. erhöhen
  LDX #$FF     ; Verzögerungsschleife
Q LDY #$30
A INY
  BNE A
  INX
  BNE Q
  JMP M        ; Nächste Verschiebung
```

Diese Routine müssen wir nur noch in das obige Achterscrolling einbauen. Für die Basic-Fans haben wir einen Basic-Lader dieses Programms am Schluß abgedruckt. Kommen wir nun zum vertikalen Scrolling:

Im Prinzip funktioniert es wie das horizontale, es wird nur ein anderes Register benutzt.

Das Register für das vertikale Scrolling ist : 53248+17.

Für das Scrolling sind wie beim horizontalen Scrolling die Bits 0 bis 2 verantwortlich.

Einzeilen Soft-Scrolling (DATA-Lader):

```
0    DATA173,22,208,41,7,201,7,208,6,32,34,128,76,0,
     128,32,68,128,238,22,208
1    DATA162,255,160,160,200,208,253,232,208,248,76,0,
     128,173,39,4,133,251
2    DATA162,38,189,0,4,232,157,0,4,202,202,224,255,
     208,243,165,251,141,0
3    DATA4,173,22,208,41,248,141,22,208,96,173,18,208,
     201,160,48,249,201,190
4    DATA16,245,96,255
10   FORI=32768TO32768+5*16
20   READA:POKEI,A
30   NEXT
40   PRINT"{CLR} HIER KOENNTE EIN TEXT STEHEN.
50   SYS32768
READY
```

(Frank Barcikowski + Holger Vocke/ah)

Sprites ohne Streß

Es soll immer noch Leute geben, die sich nicht an die Programmierung von Sprites herantrauen, weil sie von den recht unverständlichen Ausführungen im Commodore-Handbuch ziemlich verunsichert sind. Deswegen wollen wir hier die Grundlagen der Sprites darstellen.

Spielefreaks haben sie lieben gelernt: die Sprites, frei auf dem Bildschirm bewegbare Objekte in hoher grafischer Auflösung, die sogar mehrfarbig sein können. Doch nicht nur in Videospielen haben sich die Sprites bewährt: Auch als Mini-Grafik, Erkennungszeichen oder verschnörkelter Schriftzug waren sie schon zu sehen, und jede Woche entdeckt irgendein findiger Kopf eine Anwendungsmöglichkeit mehr.

Wer Sprites als grafische »Auflockerung« seiner Programme verwenden will, der kann mit dem ganz normalen Commodore-Basic hervorragende Ergebnisse erzielen, obwohl man dabei nicht um den berühmt-berüchtigten POKE-Befehl herumkommt. Wer allerdings ein rasantes Actionspiel programmieren will, der verlege sich doch besser auf Maschinensprache, denn die Bearbeitung von Sprites durch Basic ist dafür nicht schnell genug, es sei denn, man hat eine Basic-Erweiterung, die Sprites direkt unterstüzt.

Drum also hier alles, was man wissen muß, um Sprites in den verschiedensten Variationen auf die heimische Mattscheibe zu bannen.

Am Anfang ist die Idee

Wer mit Sprites arbeiten möchte, der kommt um eines nicht herum: Ein Sprite-Entwurf muß angefertigt werden. Und das bedeutet zuerst einmal, daß man kariertes Papier sowie Bunt-, Filz- oder Bleistift aus der Schublade hervorkramt. Dann malt man einen Rahmen, der genau 24 mal 21 Kästchen einschließt. Vor sich hat man nun ein sogenanntes Sprite-Entwurfsblatt liegen. Zugegeben, einige Details fehlen noch, aber zumindest können Sie damit schon einmal genau festlegen, wie Ihr Sprite später aussehen wird.

Ein vollständiges Sprite-Entwurfsblatt sehen Sie in Bild 1. Dies sollten Sie sich am besten mehrere Male fotokopieren. Für die ersten Entwürfe ist aber normales kariertes Papier empfehlenswert, weil die meistens in endlose Radiererei ausarten. Die vielen Zahlen auf dem Entwurfsblatt sollten Sie jetzt nicht verwirren: Zu deren Bedeutung kommen wir später.

Erst einmal wollen wir uns auf einfarbige Sprites beschränken. Sie können dann jedes der 504 Kästchen auf ihrem Sprite-Entwurfsblatt mit dem Stift einfärben oder auch freilassen. Jedes ausgemalte Kästchen wird später auf dem Bildschirm in der Spritefarbe zu sehen sein. Die leeren Kästchen sind später durchsichtig. Sie können also beispielsweise einen Buchstaben umrahmen, indem Sie einen Rahmen als Sprite zeichnen und dieses über den Buchstaben legen. In der Spritemitte, in der sich nur leere Kästchen befinden, ist der Buchstabe zu sehen.

So, jetzt kommt gleich zum ersten Mal der POKE-Befehl ins Spiel. Denn die nächste Frage lautet natürlich: Wie erkläre ich meinem C 64, wie das Sprite auszusehen hat? Für den Nur-Basic-Anwender, der auch umfangreiche Programme schreiben will, gibt es da nicht viele Möglichkeiten. Dem Computer muß das Aussehen des Sprites in 63 Speicherstellen mitgeteilt werden, die an bestimmten Stellen im Arbeitsspeicher stehen müssen. Wieso gerade 63? Wir erinnern uns: Ein Punkt kann entweder eingeschaltet sein oder ausgeschaltet (durchsichtig).

In einer Speicherstelle lassen sich acht Ein-/Aus-Informationen speichern, es finden also acht Punkte eines Sprites Platz. Ein Sprite ist nun 24 Punkte breit. 24 ist 3 mal 8, also benötigen wir drei Speicherstellen, um eine Zeile eines Sprites zu definieren. Schließlich hat ein Sprite noch 21 Zeilen, so daß wir 3 mal 21, das sind 63 Speicherstellen, für die Punkte eines Sprites benötigen.

Wo liegen nun diese 63 Speicherstellen? Hier haben wir, wollen wir nicht mit Unmengen von Tricks und noch mehr POKEs arbeiten, nur vier Möglichkeiten. Diese sind in Tabelle 1 aufgeführt.

Speicherbereich	Codezahl
704 — 766	11
832 — 894	13
896 — 958	14
960 — 1022	15

Tabelle 1: Speicherbereiche, in denen Spritedaten abgelegt werden können.

Und wie bekomme ich nun mein Sprite da rein, werden Sie jetzt vielleicht fragen? Nun, schauen wir uns nochmal das Entwurfsblatt an. Jetzt sollen die Zahlen entschlüsselt werden: Wie gesagt, es wird eine Zeile jeweils durch drei Zahlen verschlüsselt. Die drei dicken Striche geben an, welcher Zeilenbereich durch eine Zahl verschlüsselt wird.

Angenommen wir möchten in der ersten Reihe die Punkte 1, 3 und 5 setzen. Diese liegen alle in der ersten Spaltengruppe. Um den Zahlenwert zu errechnen, müssen Sie nun einfach die Zahlen addieren, die über den gesetzten Punkten stehen. In unserem Beispiel wären das die 128, die 32 und die 8. Das Ergebnis, 158, tragen Sie dann in das erste Kästchen mit der Überschrift »Zahlencodes« ein. Genauso verfahren Sie dann mit der zweiten und dritten Spaltengruppe und den weiteren 20 Zeilen. Schon haben Sie die 63 Zahlenwerte, die das Sprite codieren, ausgerechnet. Übrigens: Daß jede zweite Spaltentrennung gestrichelt gezeichnet ist, hat noch praktische Gründe, wie wir bei mehrfarbigen Sprites sehen werden.

Ein Sprite erscheint

Jetzt besteht also unser Sprite aus 63 Zahlen. Diese können wir nun in die angeführten Speicherbereiche POKEn (Tabelle 1). In einem Programm speichert man die 63 Zahlenwerte praktischerweise in DATAs und verwendet dann eine FOR-NEXT-Schleife, in der die Zahlenwerte in den Speicherbereich gePOKEt werden. Jedem Speicherbereich ist dabei eine Codezahl zugeordnet, die nachher noch von entscheidender Bedeutung sein wird. Die letzten drei Bereiche werden allerdings jedesmal gelöscht, wenn Sie die Datasette benutzen.

Das war aber noch längst nicht alles. Es ist noch einiges zu tun, bevor das Sprite auf dem Schirm zu sehen ist. Wir

beschränken uns, bescheiden wie wir sind, auf ein einziges Sprite, das die Nummer 0 erhält.

Als erstes sollten wir dem Video-Chip mitteilen, wo wir das Sprite im Speicher plaziert haben. Dazu dient die Speicherstelle 2040. In diese Speicherstelle POKEn Sie bitte die Codezahl des Speicherbereiches.

Dann müssen Sie das Sprite noch wie eine Lampe einschalten. Dazu POKEn Sie in die Speicherstelle 53269 einfach eine 1. Nun dürfte ihr Sprite irgendwo auf dem Schirm zu sehen sein, wahrscheinlich aber nicht da, wo Sie es gern hätten. Dem kann Abhilfe geschaffen werden: In die Speicherstelle 53248 POKEn Sie die X-Koordinate, in 53249 die Y-Koordinate der gewünschten Position. In Bild 2 finden Sie dazu eine Orientierungshilfe. Sie werden sehen, daß das Koordinatensystem für die Sprites größer ist als der Bildschirm. Ein Sprite kann also hinter dem Bildschirmrand verschwinden und ist dann nicht mehr zu sehen, obwohl es eingeschaltet ist. Ein weiteres Problem ergibt sich da: Die X-Koordinate kann laut Bild 2 bis 511 gehen. Will man aber Werte größer 255 in die Speicherstelle der X-Koordinate POKEn, erhält man einen ILLEGAL QUANTITY ERROR. Um das Sprite auch in die rechte Hälfte bewegen zu können benötigt man eine weitere Speicherstelle, 53264. Soll das Sprite auf eine Position größer als 255, dann POKEt man dort eine 1 hinein. Danach zieht man 256 von der X-Koordinate ab und POKEt das Ergebnis wieder in die Speicherstelle 53248.

Somit können wir also ein Sprite auf jede Stelle am Bildschirm und sogar über den Bildschirm hinaus positionieren.

Vielleicht gefällt ihnen die Farbe Ihres Sprites aber überhaupt nicht, vielleicht konnten Sie es bisher auch gar nicht sehen, weil es dieselbe Farbe wie der Hintergrund hatte. Nun, auch hier hilft ein POKE: Die Speicherstelle 53287 bestimmt die Farbe unseres Sprites. Dort können die 16 im Commodore-Handbuch angegebenen Farbcodes hineingeschrieben werden.

Ein Weiteres läßt sich bei Sprites kontrollieren: Die Priorität. Nehmen wir einmal an, ein Sprite und ein Buchstabe oder Grafikzeichen überschneiden sich. Dann gibt es zwei Möglichkeiten: Das Sprite ist »hinter« dem Buchstaben, wird also verdeckt, oder ist »davor«, verdeckt also den Buchstaben. Sie können einstellen, wie sich das Sprite verhalten soll: Steht in Speicherstelle 53275 eine 1, so ist das Sprite »hinten«, ist dort eine 0, ist das Sprite »vorne«.

Klein, aber fein sind diese Sprites, werden Sie jetzt wahrscheinlich sagen. Warten Sie's ab, es geht auch größer. Ein Sprite läßt sich »strecken« in X- und Y-Richtung. POKEn Sie mal eine 1 in 53277, und Sie verstehen, was damit gemeint ist. In die andere Richtung geht's mit der Speicherstelle 53271. POKEt man in beide Speicherstellen eine 0, ist der Spuk vorbei, und das Sprite hat seine »normale« Größe wieder erreicht.

Ein kleines Problem, das sogar Profis ins Grübeln gebracht hat: Streckt man ein Sprite in X-Richtung und setzt es an die X-Position Null, verschwindet es nicht vom Schirm! Die rechte Hälfte ist immer noch zu sehen. Soll nun aber das Sprite einen eleganten Abgang nach links machen, ist dies natürlich absolut störend. Doch auch hier gibt es eine bestechend einfache Lösung. Stellen Sie sich das Sprite-Koordinatensystem als Zylinder vor. Wenn Sie unser gestrecktes Sprite an die Position 511 setzen, sehen Sie wiederum fast die Hälfte des Sprites am linken Bildschirmrand. Verringert man die X-Koordinate nun langsam auf 488, wird es nach links hin entschwinden, wie wir es uns wünschen. In Y-Richtung taucht dieses Problem nicht auf.

Jetzt wird's bunt

Bisher haben wir uns auf einfarbige Sprites beschränkt. Es geht aber auch dreifarbig! Eigentlich sind es sogar vier Farben, aber die vierte Farbe ist »durchsichtig«, und wir zählen sie deswegen nicht mit.

Bitkombination	Farbe
00	Durchsichtig
11	Farbe 1
01	Farbe 2
10	Farbe 3

Tabelle [illegible]. Zusammenhang zwischen Ziffernkombinationen und Farbe bei Multicolor-Sprites.

Bevor Sie jetzt aber jubelnd die buntesten Sprites entwerfen, muß ich auf den Nachteil des »Multicolor«-Modus hinweisen: Sie haben nun nicht mehr 24, sondern nur noch 12 Punkte pro Zeile, diese sind allerdings auch doppelt so breit wie die eines tristen, einfarbigen Sprites.

Konkret bedeutet das, daß Sie schon beim Entwurf immer zwei nebeneinanderliegende Kästchen einfärben müssen, wenn Sie dort einen Punkt setzen sollen. Dabei dürfen Sie aber drei Farben benutzen. Die gestrichelten Linien in unserem Entwurfsblatt fallen also weg.

Spalten-nummer	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	Zahlencodes
Werte	128	64	32	16	8	4	2	1	128	64	32	16	8	4	2	1	128	64	32	16	8	4	2	1	
Zeile 1																									
Zeile 2																									
Zeile 3																									
Zeile 4																									
Zeile 5																									
Zeile 6																									
Zeile 7																									
Zeile 8																									
Zeile 9																									
Zeile 10																									
Zeile 11																									
Zeile 12																									
Zeile 13																									
Zeile 14																									
Zeile 15																									
Zeile 16																									
Zeile 17																									
Zeile 18																									
Zeile 19																									
Zeile 20																									
Zeile 21																									

Bild 1. Ein Sprite-Entwurfsbogen

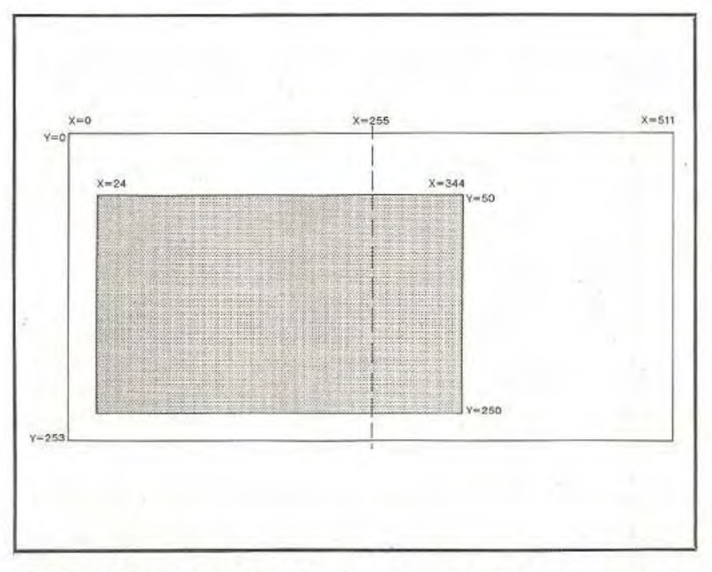


Bild 2. Mögliche Sprite-Positionen mit Bildschirmbereich

	Sprite 0	1	2	3	4	5	6	7
Codezahl-Speicher	2040	2041	2042	2043	2044	2045	2046	2047
Spritefarbe (1)	53287	53288	53289	53290	53291	53292	53293	53294
X-Position	53248	53250	53252	53254	53256	53258	53260	53262
Y-Position	53249	53251	53253	53255	53257	53259	53261	53263

Tabelle 3: Speicherstellen, die einzelne Sprites betreffen.

Beim Errechnen der 63 Zahlen für das Sprite geht man dann so vor: Anstelle einer 1 für jeden gesetzten Punkt müssen Sie nun zwei Ziffern in das Sprite-Entwurfsblatt einsetzen, eine in das linke, eine in das rechte Kästchen eines Punktes. Welche Ziffern das sind, ist von der Farbe abhängig und steht in Tabelle 2. Ab dann verläuft das Ausrechnen der Zahlenwerte des Sprites genauso wie bei den einfarbigen Sprites. Allerdings kommen noch drei zusätzliche POKEs hinzu: In die Speicherstelle 53276 muß eine 1, damit der Video-Chip weiß, daß es sich jetzt um ein Multicolor-Sprite

Einschalten	53269
Übertrag X-Position	53264
Vergrößern X	53277
Vergrößern Y	53271
Multicolor einsch.	53276
Priorität-Hintergrund	53275

Tabelle 4: Speicherstellen, die jeweils acht Schalter (je Sprite einen) enthalten.

handelt. In die Speicherstelle 53285 kommt dann die Farbe 2, in die Speicherstelle 53286 die Farbe 3. Die Farbe 1 haben wir vorher schon in der Speicherstelle 53287 festgelegt. Schon schimmert uns ein schönes mehrfarbiges Sprite entgegen, das wir dann genauso wie ein einfarbiges behandeln können.

Ein Sprite kommt selten allein

Schließlich steht ja im Commodore-Handbuch was von acht verschiedenen Sprites gleichzeitig. Also ohne viel Tricks kriegen wir nur vier verschiedene Sprites auf den Bildschirm, denn wir können ja nur vier Sprites in die Speicherstellen aus Tabelle 1 POKEn. Dafür haben wir die Möglichkeit, ein Sprite doppelt, dreifach oder sogar achtfach gleichzeitig auf den Bildschirm zu bringen. Mit acht Sprites ist dann allerdings auch die Kapazität unseres Video-Chips erschöpft.

Wie kann man nun über die sieben zusätzlichen Sprites verfügen? Diese Sprites haben praktischerweise die Nummern 1 bis 7. Für ihre Position, Farbe und ihre Codezahl haben sie ihre eigenen Speicherstellen, die in Tabelle 3 aufgeführt sind.

Komplizierter wird es da schon bei den anderen Speicherstellen, die in Tabelle 4 aufgelistet wurden. Jede dieser Speicherstellen hat Kontrolle über alle acht Sprites. Wenn Sie genauer hinschauen, bemerken Sie, daß alle diese Speicherstellen nur An-/Aus-Funktion haben, sie sind sozusagen Schalter. Nun können aber in einer Speicherstelle, wie wir wissen, acht An-/Aus-Informationen gespeichert werden, also für jedes Sprite genau eine. Jetzt folgen zwei Routinen, deren Wirkung ich Ihnen gerne erkläre, nicht aber die Funktionsweise. Dazu müßte ich längere Vorträge über Binärarithmetik, logische Funktionen und ähnliches halten und damit den Rahmen dieses Artikels sprengen.

```
POKE X, PEEK(X) OR 2 ↑ N
```

legt den Schalter in der Speicherstelle X für das Sprite N auf »An« um, ohne die anderen Schalter zu beeinflussen.

```
POKE X, PEEK(X) AND (255 - 2 ↑ N)
```

stellt den entsprechenden Schalter wieder auf »Aus« um, ohne die anderen zu beinflussen.

Wichtig sind die beiden Adressen in der Tabelle 5: Sie sind uns schon bekannt, und legen die Farben 2 und 3 für **alle** Sprites verbindlich fest. Zwei Sprites im Multicolormodus, die gleichzeitig erscheinen, können sich also nur in der Farbe 1 unterscheiden.

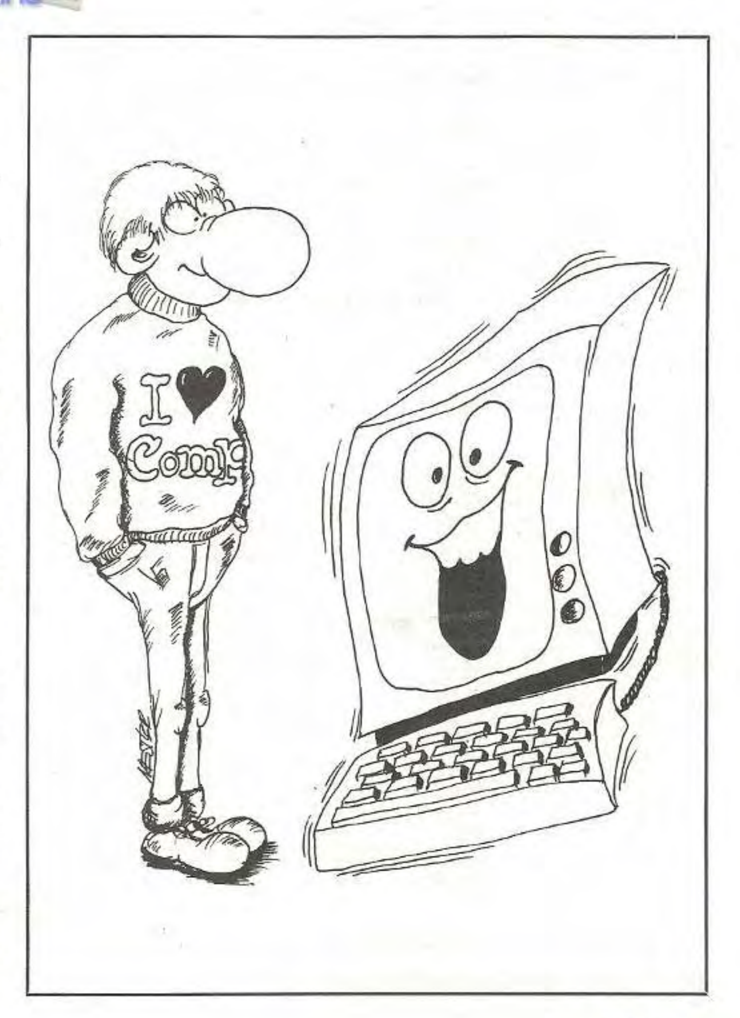
Farbe 2	53285
Farbe 3	53286

Tabelle 5: Speicherstellen, die auf alle acht Sprites gleichzeitig wirken

Damit ist das Wichtigste über Sprites gesagt. In Basic sollte man sich auf stehende Sprites beschränken. Sich bewegende Sprites sind relativ langsam, wenn das Basic-Programm auch andere Funktionen nebenbei erfüllen muß.

Wer Sprites öfters benutzen will, für den lohnt es sich, die ebenfalls in diesem Heft abgedruckte Basic-Erweiterung »Sprite-Basic« abzutippen, und das soeben Gelernte dann nicht mit POKEs, sondern mit richtigen Befehlen auszuführen. Denn dann geht der Sprite-Spaß erst richtig los!

(Boris Schneider/ev)



Hardcopy für Ihren Drucker

... oder: von den Problemen, den Bildschirminhalt auf Papier zu bringen — und natürlich von den Lösungen dieser Probleme!

Das Problem dürfte sich jedem Benutzer des C 64 stellen, der einen Drucker hat: Auf dem Monitor oder Fernseher wird ein Bild dargestellt, doch leider ist es ganz unmöglich, dieses Bild auf dem Drucker auszugeben, denn es wird nicht der normale Bildschirm mit einer Auflösung von 25 Zeilen á 40 Zeichen dargestellt, sondern ein Bild in hochauflösender Grafik. Wie Sie trotzdem auch mit diesem Problem fertig werden, soll im folgenden beschrieben werden.

Aller Laster Anfang: Ihr Drucker

Schon beim Drucker stoßen wir auf schwerwiegende Probleme, denn manche Drucker sind wirklich nur sehr schwer oder gar nicht zum Drucken von Grafik zu bewegen. Folgende Kriterien sollten bei Ihrem Drucker erfüllt sein:

— der Drucker ist grafikfähig (natürlich die wichtigste Voraussetzung);

— der Drucker druckt mit mindestens acht Nadeln oder ähnlichem. Damit ist gemeint, daß der Druckkopf so konstruiert ist, daß er vertikal acht Punkte untereinander drucken kann;

— der Drucker beherrscht den sogenannten BI-Mode (Bit-Image-Mode). Das ist der Modus, in dem die hochauflösende Grafik ausgedruckt wird.

Wenn Ihr Drucker diese Voraussetzungen erfüllt, dann ist es Ihnen in jedem Fall möglich, Ihre Bildschirmgrafiken auf Papier zu bringen.

Aber auch Drucker, die keinen BI-Modus haben, sind mitunter grafikfähig. Da die Druckerpalette jedoch sehr groß ist, werden wir auf keinen speziellen Drucker eingehen, und es bleibt Ihnen letztlich überlassen, die Tauglichkeit Ihres Druckers zu überprüfen.

Auch bei Druckern der unteren Preisklasse findet man heute schon die sogenannten ESC-Sequenzen. Das sind Steuerbefehle für den Drucker, die ihn zum Beispiel dazu veranlassen, die Schriftart oder etwa den Zeilenvorschub zu verändern. Eine solche Steuersequenz erkennt der Drucker an einem speziellen Zeichencode, bei dem er nicht wie sonst üblich das empfangene Zeichen als solches ausgibt, sondern vielmehr auf weitere Zeichen wartet, die in ihrer Gesamtheit einen Steuerbefehl ergeben. Üblicherweise wird dabei als ESC-Code das Zeichen CHR$(27) an den Drucker gesendet.

ESC — (von escape, engl. für entkommen, flüchten)

Die ESC-Sequenzen sind leider in keiner Norm vereinheitlicht. Deswegen darf normalerweise die Kompatibilität zwischen Druckern gleicher Leistungsklasse nicht ohne Einschränkungen angenommen werden.

Nach Senden von ESC an den Drucker werden immer Daten nachgeschoben, die die Grundeinstellung des Druckers ändern, so etwa den Drucker dazu veranlassen, Grafik anstatt Klartext zu drucken. Wenn Sie einmal Ihr Druckerhandbuch aufschlagen und eine der Sequenzen ESC-K, ESC-L oder ESC-Z finden, dann werden Sie (hoffentlich) feststellen, daß Sie in dem Teil des Handbuchs lesen, der Sie interessiert — der Ausgabe von Grafik.

Jetzt sind noch ein paar Berechnungen anzustellen und schon kann das Bild auf dem Drucker ausgegeben werden. Doch diese Berechnungen haben es in sich!

Wie die Grafik im Computer steht

Der C 64 ist bekanntlich ein vielseitiger Computer. So hat er unter anderem in seinem Inneren einen kleinen Chip (6567 Video Interface Chip, kurz VIC II) eingebaut, der dafür verantwortlich ist, daß sich etwas auf Ihrem Bildschirm tut. Der VIC II ist sehr flexibel und hat vielseitige Verwendungsmöglichkeiten innerhalb des C 64; wir beschränken uns jedoch hier auf die Funktionen, die ausschließlich für die Erzeugung der Grafik verantwortlich sind.

Es können acht Grafikbildschirme vom VIC II verwaltet werden; ihre Basisadressen sind ein Vielfaches von 8192. So beginnt etwa der dritte Grafikbildschirm bei der Basisadresse 16384 (bei 0*8192 liegt der erste Grafikschirm, dieser wird aber im Regelfall nicht benutzt, da der Bereich anderweitig belegt ist, zum Beispiel durch Zero-Page und Stack.

Wenn Sie ein Bild drucken wollen, müssen Sie sich immer erst Klarheit darüber verschaffen, in welchem Bereich der Grafikbildschirm steht, den Sie ausgeben wollen. Zwar können Sie für die Berechnungen auch so lange zwischen den

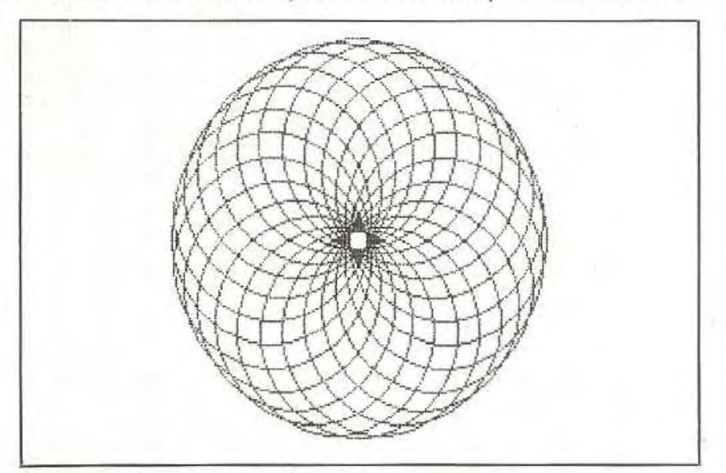


Bild 1. Beispiel einer HiRes-Darstellung am Commodore 64

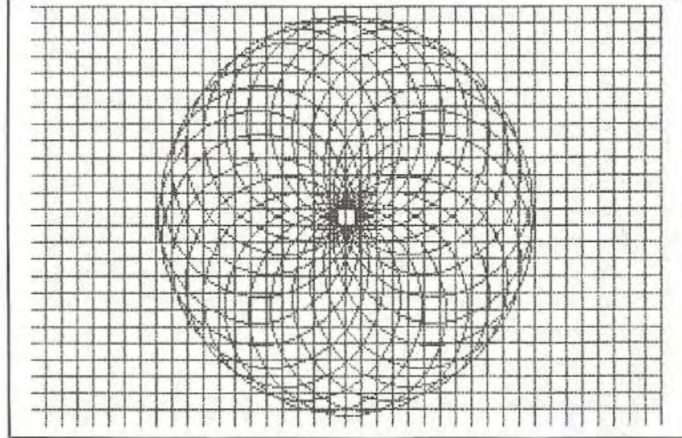


Bild 2. Die Rasterung verdeutlicht den Aufbau des HiRes-Bildschirms

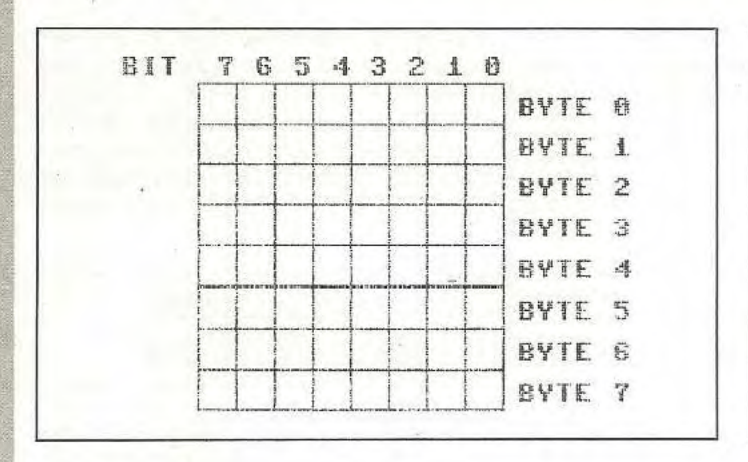


Bild 3. Aufbau einer einzelnen Rasterzelle aus Bild 2

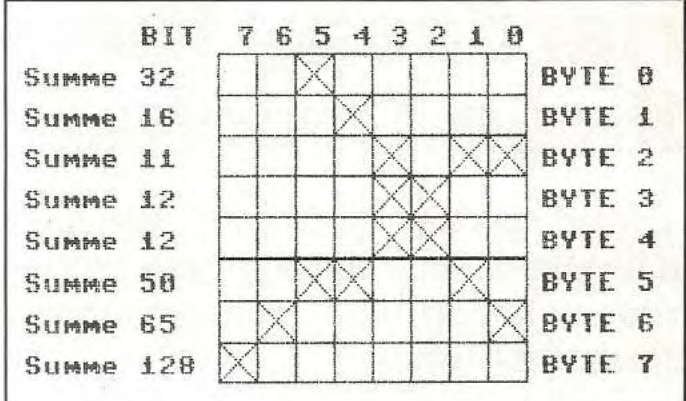


Bild 4. Beispiel zur Berechnung der Zeilen-Wertigkeiten innerhalb einer HiRes-Rasterzelle

acht Basisadressen wechseln, bis Sie den richtigen Bildschirm erwischt haben, effektiver dürfte da aber das kleine Basic-Programm sein, das in Listing 1 dargestellt ist. Mit Hilfe dieses kurzen Programms werden die Grafikbildschirme 3 (Startadresse 16384) bis 8 (Startadresse 57344) durch ein kurzes Maschinenprogramm auf die Startadresse des zweiten Grafikschirms verschoben und dort dargestellt. Dadurch wird es Ihnen möglich, auch die Grafikbildschirme 5 und 7 von Basic aus problemlos auszugeben, da diese Schirme normalerweise nicht erreichbar sind (sie liegen »unter« dem ROM). Weil sich dieser Artikel nur mit der Ausgabe von Grafik beschäftigt, verzichte ich an dieser Stelle auf Erläuterungen zu dem Programm, denn dabei würde zu stark auf die Besonderheiten des VIC II eingegangen werden müssen.

Nicht genug der acht Grafikschirme, verfügt der VIC II auch noch über zwei verschiedene Darstellungsarten: den sogenannten »High Resolution Mode« (HiRes, hochauflösende Grafik mit 64000 einzelnen Bildpunkten) und den »Multicolor-Mode«, in dem jeder Einzelpunkt in vier verschiedenen Farben dargestellt werden kann, bei dem aber dafür die Auflösung nur noch 32000 Bildpunkte beträgt. So mancher Anwender hat sich schon darüber gewundert, daß seine Hardcopy-Routine nur ein seltsames Strichmuster aufs Papier gebracht hat, aus dem man mit etwas Fantasie andeutungsweise das Originalbild entziffern konnte. Der Grund: im Multicolor-Modus werden in den Daten nicht nur die Bildinformation, sondern auch die Farbinformation verschlüsselt.

An dieser Stelle ein Tip: wenn Sie eine Grafik ausgeben wollen, jedoch keine Möglichkeiten haben, den Programmablauf zu unterbrechen, so können Sie durchaus einen Reset auslösen. Dabei wird das Bild normalerweise nicht gelöscht und Sie können dann fast immer von Basic aus weitere Schritte zur Ausgabe des Bildes einleiten.

High Resolution Mode

In Bild 1 ist zunächst einmal ein HiRes-Bild als solches dargestellt. In Bild 2 wurde über das Bild zusätzlich eine Schraffur gelegt, um die Struktur der Ablage des Bildes zu verdeutlichen. Durch das Rastermuster wird angedeutet, daß die Informationen nicht etwa Reihe für Reihe abgelegt werden, sondern in kleinen Blöcken, von denen in eine Zeile 40 passen, vertikal dagegen nur 25. Der Fachmann wird sich sofort an die Normaldarstellung von Zeichen erinnern. Tatsächlich ist diese Darstellung eng mit der normalen Zeichendarstellung verwandt.

Bild 3 zeigt eine Vergrößerung eines Blocks. Man erkennt, daß ein Block seinerseits aus einer Matrix von 8 x 8 Punkten besteht. Ergo besteht ein Block aus 64 Punkten, und wir wissen, daß 25 x 40 = 1000 dieser Blöcke auf dem Bildschirm Platz haben, wodurch also insgesamt eine Auflösung von 64000 Punkten erreicht wird. So weit, so gut. Aus Bild 3 geht aber noch weiter hervor, daß die Informationen innerhalb eines einzelnen Blocks zeilenweise abgelegt werden; ein Block setzt sich also aus 8 Bytes zusammen, die ihrerseits wieder in kleinste Informationseinheiten, die Bits, zerlegt werden können. Diese Blöcke werden bei der HiRes-Darstellung so lange aneinandergereiht, bis eine Zeile mit Blöcken (weiterhin auch als »Blockzeile« bezeichnet) gefüllt ist. Der nächste Block liegt dann naheliegenderweise unter dem ersten der vorangegangenen Zeile.

Damit es etwas anschaulicher wird, wollen wir uns an dieser Stelle mit einem kleinen Beispiel befassen.

Aus Bild 2 ist Block 27 in Zeile 13 vergrößert im Bild 4 wiedergegeben (Blockbereich 0-39, Zeilenbereich 0-24). Weiterhin wird für die Beispielrechnung angenommen, daß das Bild im Computer im Grafikschirm 2 (Basisadresse 8192) abgespeichert ist. Wenn Sie nun die Blöcke zeilenweise auszählen, werden Sie feststellen, daß der besagte Block die Nummer 547 (erster Block entspricht der Nummer 0) trägt.

Zunächst wollen wir klären, wie der Inhalt gerade dieses Blocks zu ermitteln ist. Dazu berechnen wir beispielsweise die Adresse von Byte 0 innerhalb dieses Blocks nach der Formel

```
Byte 0 = Basisadresse+Blockzeilen*320+Blocknummer*8
```

Wie läßt sich die Formel erklären?

Zunächst einmal ist es verständlich, daß die Basisadresse des Grafikschirms der Positionsberechnung zugrunde liegen muß. Weiterhin faßt eine Blockzeile 40 Blöcke zu je 8 Byte; eine Blockzeile besteht also aus 320 Byte. Nun muß man sich noch innerhalb der aktuellen Zeile zu dem Block bewegen, dessen Inhalt untersucht werden soll. Hierzu multipliziert man einfach die Anzahl der Blöcke mit 8, da sich ein Block aus 8 Byte zusammensetzt. Beachten Sie bitte bei der Berechnung, daß die Blockzeilen im Bereich 0 bis 24 liegen, die Blocknummern dagegen im Bereich 0 bis 39. So ergibt sich zum Beispiel als Startadresse für den 27. Block in der 13. Zeile:

```
Byte 0 = 8192 + 13 * 320 + 27 * 8 = 12568
```

Bei dem Block in unserem Beispiel ergibt also der Basic-Befehl

```
PRINT PEEK(12568)
```

den Inhalt von Byte 0 des Blocks 547. Wie groß wird nun dieser Wert sein?

Wie bereits erwähnt, besteht ein Byte aus kleinen Informationseinheiten (Bits). Das Bit kann entweder den Zustand »1« oder den Zustand »0« annehmen. Wenn Sie nun noch einmal Bild 3 betrachten, werden Sie erkennen, daß 8

solcher Bits ein Byte ergeben. Diese Bits werden mit Bit 0 bis Bit 7 von rechts nach links durchnumeriert. Da das Bit nur zwei Schaltzustände annehmen kann, wird zur Berechnung eines Bytes das Binärsystem zugrunde gelegt. Es ist

$2^7=128$
$2^6=\ 64$
$2^5=\ 32$
$2^4=\ 16$
$2^3=\ \ 8$
$2^2=\ \ 4$
$2^1=\ \ 2$

und schließlich

$2^0=\ \ 1$

Bereits mit diesen Vorüberlegungen läßt sich der Inhalt von Byte 0 berechnen. Wie Sie in Bild 4 sehen, ist nur Bit 5 gesetzt, also ergibt sich als Inhalt von Byte 0 der Wert $2^5 = 32$. Diesen Wert würde auch die Abfrage »PRINT PEEK(12568)« ergeben.

Entsprechend ergeben sich zum Beispiel für Byte 2 als Inhalt

$2^3 + 2^1 + 2^0 = 8 + 2 + 1 = 11.$

Damit lassen sich auch alle anderen Bytes eines Blocks berechnen. Das Verfahren verläuft analog zum gezeigten Beispiel. Die Inhalte aller Bytes des Beispiels sind vollständig in Bild 4 wiedergegeben.

Der Drucker kommt ins Spiel

Zunächst muß der Drucker für die Datenausgabe vorbereitet werden. Hierfür müssen Sie sicherstellen, daß ein eventuell vorhandenes Interface keine falsche Interpretation von gesendeten Daten vornimmt. Dies geschieht im Normalfall durch Setzen einer bestimmten Sekundäradresse, bei einigen Herstellern auch als »Linearkanal« bezeichnet, weil gesendete Zeichen im Original an den Drucker weitergeleitet werden.

Ebenso muß der Zeilenvorschub korrigiert werden. Wenn man davon ausgeht, daß acht Punktreihen untereinander in einem Arbeitsgang gedruckt werden können, so ist der Zeilenvorschub so einzustellen, daß die nächste gedruckte Zeile sich nahtlos an die vorhergehende Zeile anschließt. Dazu ist der Zeilenvorschub im Regelfalle auf 24/216 Zoll zu stellen.

Damit sind die besonderen Voreinstellungen, die sowohl für den HiRes- als auch den Multicolor-Modus Gültigkeit besitzen, vorgenommen. Weitere Operationen beziehen sich nun auf den jeweiligen Modus, von dem ausgegangen wird.

Kommen wir schließlich zu dem schon anfangs erwähnten ESC-K und ESC-L. Mit diesen Sequenzen wird der BI-Mode eingeschaltet. Doch das reine Einschalten genügt nicht; man muß auch noch spezifizieren, wieviele Daten zum Drucker gesendet werden.

Schauen wir uns zunächst an, wie Grafik auf dem Drucker ausgegeben wird. Als Beispiel nehmen wir wieder Bezug auf Bild 4. Im ersten Teil haben wir überlegt, wie ein Block innerhalb des Computers abgelegt wird: ein Block besteht aus 8 Bytes, die horizontal untereinander stehen.

Beim Drucker ist das nicht üblich. Da der Druckkopf im allgemeinen nur aus einer Reihe von vertikal untereinander liegenden Dots (Nadeln) besteht, werden die Daten vertikal aufbereitet erwartet. Wird deshalb im Grafikmodus des Druckers ein Byte gesendet, so werden maximal 8 Punkte vertikal untereinander gedruckt. Wir können vom Computer her jedoch immer nur horizontale Punktreihen per PEEK abfragen. Was benötigt wird, ist eine Routine, die jeweils einen Block aus 8 x 8 Punkten entsprechend umrechnet. Bei unserem Beispiel muß der Block aus Bild 4 entsprechend Bild 5 umgerechnet werden. Bei der Berechnung gehen wir in diesem Fall von der Tatsache aus, daß das höchstwertige Bit der Druckerkopfdots innerhalb der Druckkopfmatrix oben liegt. In Ihrem Handbuch kann aber auch nachzulesen sein, daß das höchstwertige Bit in der Druckmatrix nach unten weist. In diesem Fall würden sich die berechneten Werte gemäß Bild 6 ändern. Wie Sie sehen, ist es durchaus wichtig zu wissen, wie die Dotmatrix Ihres Druckers aufgebaut ist.

Wir gehen davon aus, daß der zuerst beschriebene Fall zutrifft. Wie man relativ einfach die Matrix von Basic aus umrechnen kann, wird durch das Basic-Programm in Listing 2 illustriert. Mit diesem kleinen Programm können Sie jede beliebige Matrix umrechnen lassen. Dazu geben Sie nur die Startadresse Ihres Grafikschirms, die Zeile, in der sich der Block befindet und schließlich die Nummer des Blocks ähnlich wie oben in der Formel erwähnt an. Der Inhalt des betreffenden Blocks wird in dem Basic-Programm in Listing 2 in den Zeilen 190 bis 210 in das Feld BYTE eingelesen. Nun müssen die Daten gewandelt werden. Dazu folgende Überlegungen:

Um das erste Bit links oben des ersten vertikalen Bytes zu erhalten, muß von Byte 0 Bit 7 betrachtet werden (siehe Bild 4). Wenn es gesetzt ist, so wird es mit $2^7 = 128$ multipliziert; man erhält so einen ersten Wert. Dann wird von Byte 1 Bit 7 betrachtet und untersucht, ob das Bit gesetzt ist. Falls dies der Fall ist, wird es mit $2^6 = 64$ multipliziert und zum ersten Wert addiert. Dann wird jeweils nach dem gleichen Schema

BYTE 0 1 2 3 4 5 6 7
BIT 7
BIT 6
BIT 5
BIT 4
BIT 3
BIT 2
BIT 1
BIT 0
Summe 1 2 132 68 56 24 36 34

Bild 5. Das gleiche Zeichen wie in Bild 4, aber zur Druckeransteuerung um 90 Grad transformiert

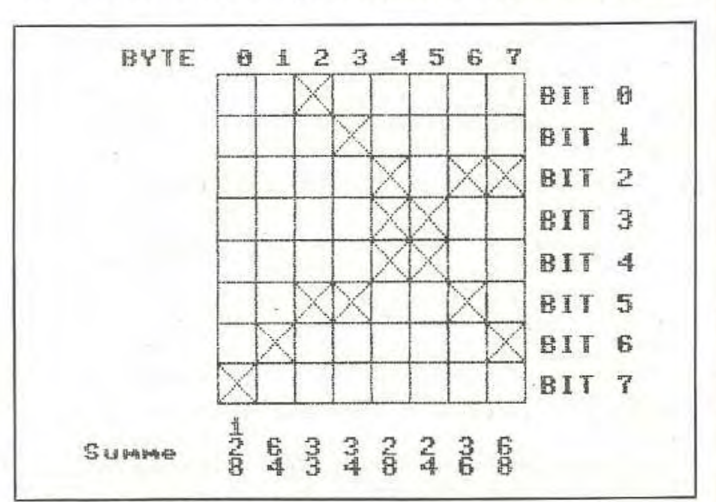


Bild 6. Wie Bild 5, aber für Drucker mit entgegengesetzter Nadelnumerierung (Bit 7 unten)

von Byte 2 bis 7 das 7. Bit untersucht und schließlich mit einer absteigenden Potenz von 2 multipliziert und zum jeweils schon erhaltenen Wert (im Listing 2 die Variable OUTPUT) hinzuaddiert. Hat man auf diese Art und Weise das erste vertikale Byte erhalten, kann selbiges an einen Drucker gesendet werden, denn jetzt ist die gewünschte Umwandlung vorgenommen worden, wovon Sie sich durch Nachrechnen überzeugen können.

Da der Block aber aus 8 Bytes besteht, wird dann mit der Aufbereitung des 2. vertikalen Bytes fortgefahren, also Bit 6 von Byte 0 untersucht. Ist es gesetzt, so muß es wieder mit 2^7 multipliziert werden, damit es an die korrekte Stelle innerhalb der Dotmatrix rutscht. Wieder erhält man einen ersten Zwischenwert. Dann wird Byte 1 auf Bit 6 hin untersucht. Ist es gesetzt, dann wird es mit 2^6 multipliziert, um dieses Bit an die korrekte Stelle der Dotmatrix des Druckers zu bringen. Nach diesem Verfahren wird fortgefahren, bis auch das letzte vertikale Byte berechnet worden ist.

Wenn Sie das Programm also starten und die Adresse eines Blocks eingeben, so erhalten Sie acht Werte, die jeweils (vergleiche hierzu nochmals den Zusammenhang zwischen den Bildern 4 und 5) die umgerechneten Werte wiedergeben.

Um die Effektivität und Berechnungszeit zu kürzen, wurde die Umrechnung in einen sehr kompakten Ausdruck eingebunden, über den Sie nicht gleich verzweifeln sollten (Zeile 240). Der Ausdruck kann nur verstanden werden, wenn Sie sich schon mal mit der »And/Or-Boolean-Wüste« des Computers beschäftigt haben, und ich möchte deswegen auch hier auf tiefergehende Erklärungen des Ausdrucks verzichten. Wenn Sie sich trotzdem für die Thematik interessieren, so empfehle ich Ihnen die im 64'er, Ausgabe 7/85 gestartete Serie »Logeleien«. Wichtig ist hier aber nur, daß Sie nachvollziehen können, wie die Umrechnung im Prinzip funktioniert.

Block an Block — Reihe an Reihe — fertig ist das HiRes-Bild

Da nun ein Programm erstellt ist, das einzelne Blöcke gemäß unseren Anforderungen aufbereitet, sind wir fast an unserem Ziel angelangt. Die Drucker, bei denen es nicht möglich ist, mehr als einen Block auf einmal zu senden, haben praktisch damit das endgültige Programm, mit dem der Druck von Grafik möglich gemacht wird. Besser dran sind da schon die Besitzer von Druckern, die es erlauben, Grafik zeilenweise auszugeben. Was jetzt nur noch fehlt, sind Schleifen, die die Blöcke hintereinander berechnen und eine korrekte Aufbereitung vornehmen.

```
Basis des Grafikschirms eingeben
Zeilenvorschub auf 24/216''
    Sende Code zur Einschaltung von BIM
        Block berechnen und ausgeben
    bis alle Bloecke einer Zeile fertig
    Sende Zeilenvorschub
bis die Zeilen 0- 24 durchlaufen sind
Zeilenvorschub auf 1/6'' einstellen
```

Bild 7. Struktogramm zur Erstellung einer HiRes-Hardcopy

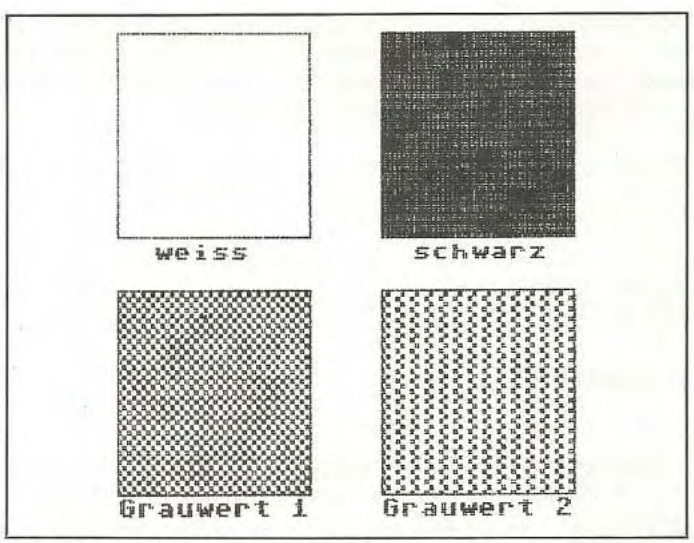


Bild 8. Verschiedene Graustufen für eine Multicolor-Hardcopy

Doch bevor die eigentliche Ausgabe beginnt: der Drucker muß bereit sein! Haben Sie also den Zeilenvorschub korrigiert?

Nun, dann dürfte Ihr Programm dem Struktogramm in Bild 7 entsprechen.

Das Struktogramm enthält einen Punkt, über den man sich noch Klarheit verschaffen sollte:

»Sende Code zur Einschaltung von HiRes«.

Hier stellt sich die Frage, was man dem Drucker mitteilen soll. Da die Daten einer Blockzeile hintereinandergereiht auch einer Blockreihe auf dem Drucker entsprechen sollen, wird deutlich, was dem Drucker mitgeteilt werden muß:

— BI-Mode einschalten, dabei geeignete Punktdichte wählen

— Anzahl der Daten definieren

Da 40 Blöcke zu je 8 Byte gesendet werden sollen, muß dem Drucker mitgeteilt werden, daß 320 Zeichendaten folgen. Üblicherweise wird dazu zur Einschaltung des BI-Mode ESC-L beziehungsweise ESC-K verwendet. Da die Vorgehensweise bei beiden Steuerbefehlen gleich ist und sie sich nur bezüglich Ihrer Punktdichten unterscheiden, betrachten wir nur ESC-L. Sie werden in Ihrem Druckerhandbuch finden, daß nach dem Einschalten von ESC-L noch definiert werden muß, wieviele Zeichendaten gesendet werden, bevor der Drucker wieder in seinen normalen Modus zurückschaltet. Würde nämlich keine solche Definition vorhanden sein, so würde der Drucker jedes gesendete Byte nach Einschaltung des BI-Mode als Grafikzeichen interpretieren und dann wäre zum Beispiel ein kontrollierter Zeilenvorschub unmöglich zu realisieren. Da Sie inzwischen wissen, daß sich eine Blockzeile im Computer und deshalb entsprechend auch später auf dem Drucker aus 320 Bytes zusammensetzt, müssen Sie vor der eigentlichen Ausgabe der Blockzeile dem Drucker die nachfolgende Anzahl von Zeichendaten mitteilen. Dazu werden üblicherweise zwei Zahlen n1 und n2 nach den Formeln

```
n1 = Zahl der Daten - INT(Zahl der Daten/256) * 256
```

und

```
n2 = INT(Zahl der Daten/256)
```

berechnet. In unserem Beispiel wären also n1 = 64 und n2 = 1. An den Drucker würde man also eine Steuersequenz senden, die folgendermaßen aussieht:

```
CHR$(27);"K";CHR$(64);CHR$(1);
```

Danach ist dann der Drucker im BI-Mode und erwartet 320 Zeichendaten, um dann wieder auf Normalbetrieb zurückzuschalten.

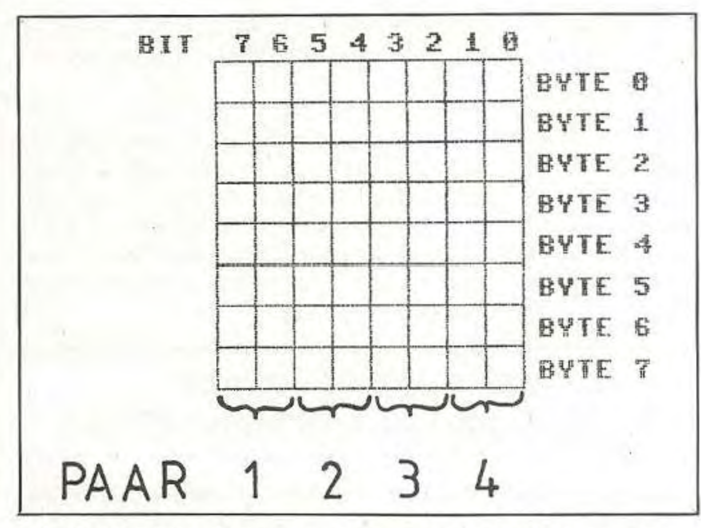


Bild 9. Die Aufteilung einer Rasterzelle in Bitpaare bei Multicolor

Das entsprechende Basic-Programm für die Ausgabe eines ganzen Grafikschirms im HiRes-Modus ist in Listing 3 dargestellt. Beachten Sie aber Ihre druckerspezifischen Einschränkungen, bevor Sie das Programm ausprobieren. Überprüfen Sie bitte, wie bei Ihrem Drucker die einzelnen Funktionen angesprochen werden. Hilfestellung sollten Ihnen dabei das Struktogramm aus Bild 7 und das eigentliche Basic-Programm mit den reichlich eingefügten REM-Statements in Listing 3 bieten. Wenn Sie das Listing 2 mit Listing 3 vergleichen, werden Sie weiterhin feststellen, daß der Algorithmus zur Umrechnung eines Blocks übernommen wurde. Mit diesen Abschlußbemerkungen wenden wir uns jetzt dem zweiten Problem zu: dem Druck von Grafik im zweiten Grafikmodus.

Der zweite Akt: Multicolor-Modus

Der Multicolor-Modus unterscheidet sich erheblich vom HiRes-Modus. Das liegt daran, daß mehrere Farben verwendet werden dürfen. Dabei werden immer zwei hintereinanderliegende Punkte zu einer Informationseinheit zusammengefaßt, aus der sich ableiten läßt, ob ein Punkt gesetzt ist, und wenn, in welcher Farbe. Dargestellt werden können in dieser Betriebsart des VIC II nur noch 32000 Bildpunkte.

Ein Punktpaar kann folgende Schaltzustände annehmen (1 = Bit gesetzt, 0 = Bit gelöscht):

00 Punkt nicht gesetzt, wird in Hintergrundfarbe dargestellt
01 Punkt gesetzt, Darstellung in Farbe 1
10 Punkt gesetzt, Darstellung in Farbe 2
11 Punkt gesetzt, Darstellung in Farbe 3

Wie man sich leicht selbst überlegt, muß eine Hardcopy einer Grafik dieses Modus also nicht nur zwischen gesetzten und nicht gesetzten Punkten differenzieren, sondern auch noch zusätzlich die Farbinformation weitergeben.

Drei Farben auf dem Papier — ist das möglich?

Mit einem normalen Drucker wird es selbstverständlich nicht möglich sein, Farben darzustellen. Der Drucker ist nur in der Lage, schwarze Farbe auf das Papier zu bringen. Es drängt sich hier der Vergleich mit einem Schwarzweiß-Fernsehgerät auf. Dort werden die unterschiedlichen Farben durch entsprechende Grauschattierungen wiedergegeben, die aus Mischprodukten von Weiß und Schwarz entstehen.

Das gleiche Prinzip läßt sich auch auf dem Drucker verwirklichen. Dabei wählt man unterschiedliche Punktdichten, um so den Eindruck zu vermitteln, daß das Bild aus unterschiedlichen Farben besteht. Für uns reicht es wegen der maximal vier darstellbaren Farben aus, anzunehmen, daß die Farben Weiß (Hintergrundfarbe), Schwarz, und zwei Graustufen bereits das ganze Bild darstellen können. Bild 8 zeigt verschiedene Rastermuster.

So steht ein Multicolor-Bild im Speicher

Wieder wird über die Grafik ein Raster gelegt, welches das ganze Bild in einzelne Blöcke aufteilt. Eine Blockreihe besteht ebenfalls wieder aus 40 Blöcken. Ein Block ist in Bild 9 vergrößert. Zu erkennen ist die Struktur der Bitpaare, so daß im Prinzip eine Matrix vorliegt, mit der 4 Punkte in X-Richtung und 8 Punkte in Y-Richtung dargestellt werden können. Sie erhalten also eine 4 x 8-Matrix, die dann allerdings auch Informationen zur Farbe der einzelnen Punkte enthält. Zu finden ist nun ein Weg, der wieder Blöcke auswertet und zusätzlich noch Farbinformationen in Form eines Rastermusters mitliefert.

Wie kann man nun die Informationen, die der Block beinhaltet, verarbeiten?

Betrachten Sie zum Beispiel das fünfte Byte des Blocks in Bild 4. In Multicolor-Darstellung setzt sich das Byte aus 4 Bitpaaren zusammen, von denen das erste (Bitpaar 00) mit der Farbe des Hintergrundes, das zweite (Bitpaar 11) in Farbe 3, das dritte wieder in der Hintergrundfarbe und das vierte (Bitpaar 10) in Farbe 2 dargestellt werden würde. Ein Programm muß nun dem Inhalt eines Bitpaares verschiedene Punktdichten zuordnen, die dann als Graustufen auf dem Papier wiedergegeben werden.

Graustufen mit dem Drucker

Wenn davon ausgegangen wird, daß jedes Byte eines Blocks getrennt für sich ausgewertet wird, stellt sich einerseits der Sachverhalt relativ einfach dar, andererseits muß aber mit entsprechender Berechnungszeit kalkuliert werden.

Um ein gutes, kontrastreiches Bild entstehen zu lassen, muß die Punktdichte möglichst groß gewählt werden. Grafikfähige Drucker bieten hierzu meist einen BI-Mode »vierfache Dichte« an, mit dem eine genügend hohe Auflösung erzielt werden kann. Um das Bild nun möglichst im Maßstab 1:1 auf dem Drucker auszugeben, muß ein gesetzter Punkt auf dem Drucker durch ein kleines Quadrat wiedergegeben werden.

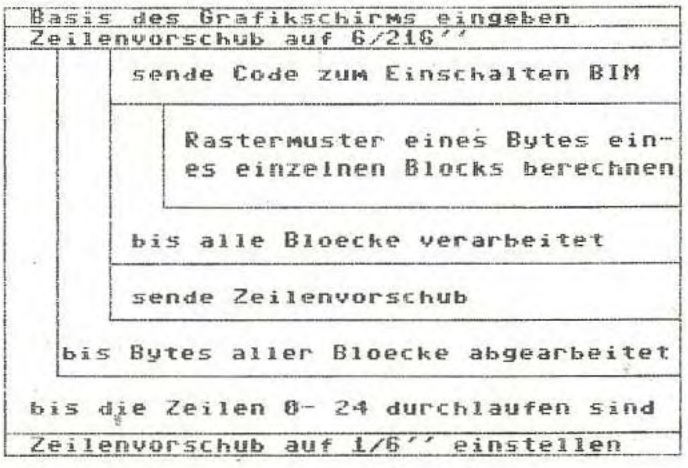


Bild 10. Struktogramm zur Erstellung einer Multicolor-Hardcopy

Bild 11. Beispielausdruck in »Multicolor«

Indem innerhalb eines solchen einzelnen Quadrates unterschiedliche Punkte gesetzt werden, erhält man unterschiedliche Grauwerte. Betrachten Sie hierzu einmal das Programm in Listing 4. Ähnlich wie im Fall einer HiRes-Grafik beschäftigen wir uns mit der Umwandlung eines einzelnen Blocks. Die Bestimmungsformeln zur Positionsbestimmung eines Blocks sind identisch mit denen zur Positionsbestimmung eines HiRes-Blocks, die Auswertung dagegen erfolgt anders. Die Rastermuster in den Zeilen 160 bis 190 erzeugen bei der gewählten vierfachen Punktdichte bei Ausgabe auf dem Drucker jeweils ein kleines Quadrat, das einen gesetzten Punkt in entsprechendem Grauwert wiedergibt (weiß wird hier auch mit einem Grauwert identifiziert). Diese Rastermuster werden zu Beginn des Programms in ein Feld DRUCKMATRIX eingelesen. Danach wird wie üblich definiert, welcher Block des Grafikbildschirms ausgegeben werden soll.

Während bei einer HiRes-Grafik die vertikalen Werte eines Blocks auf dem Bildschirm ausgegeben wurden, wird hier dagegen direkt der Drucker angesprochen, um das Prinzip der Umwandlung der Bitpaare in Rastermuster zu verdeutlichen. Da die Informationen innerhalb eines Blocks byteweise verarbeitet werden und da natürlich nach dem ersten berechneten Rastermuster für Byte 0 das zweite Rastermuster direkt unter dem ersten liegt, muß zusätzlich der Zeilenvorschub entsprechend korrigiert werden. Dazu wird der Zeilenvorschub auf zirka 6/216 Zoll gesetzt (siehe Zeile 270). Da weiterhin ein Byte des betrachteten Blocks aus vier Bitpaaren besteht, müssen selbstverständlich auch vier Rastermuster nebeneinander auf dem Drucker erzeugt werden.

Das Bit-Rastermuster

Weil das Rastermuster eines Bits aber schon aus acht Daten (siehe DATA-Zeilen) zusammengesetzt wird, werden also 32 Daten pro Byte an den Drucker gesendet. Daher erklärt sich die Sequenz CHR$(32);CHR$(0) in Zeile 300, durch die der BI-Mode eingeschaltet wird und 32 Daten als reine Grafikdaten vom Drucker erwartet werden. Dann wird in den Zeilen 330 der Inhalt eines Bytes eines Blocks ermittelt und daraufhin in den Zeilen 340 bis 380 in seine Bitpaare zerlegt.

Entsprechend dieser Zerlegung erhält die Variable RASTER einen Wert zwischen null und drei (wegen logischer Verknüpfungsoperationen). Durch die so zustande kommenden Inhalte der Variable RASTER kann an den Drucker die passende Druckmatrix gesendet werden. Mit Byte 1 bis 7 des gleichen Blocks wird genauso verfahren. Sie erhalten dadurch auf dem Drucker ein Rastermuster, dessen Aussehen von den Eingangsdaten abhängt.

Der nächste Schritt: Ausgabe eines Multicolor-Bildes

Ein Block besteht aus 8 untereinanderliegenden Bytes. Von diesen Blöcken stehen in einer Zeile 40 und ergeben so eine Blockreihe. Wenn wir nach dem oben beschriebenen Prinzip weiter vorgehen wollen, dann muß zunächst Byte 0 von Block 1 berechnet werden. Daraufhin darf man jedoch nicht das zweite Byte des Blocks auswerten, sondern muß Byte 0 des Blocks 2 berechnen, da die nächste Information neben dem Byte 0 des Blocks 1 liegt. Man fährt nach diesem Auswertungsprinzip fort, bis Byte 0 des Blocks 40 einer Blockreihe berechnet ist und führt erst dann einen Zeilenvorschub aus, der aber so bemessen sein muß, daß die folgenden Punkte direkt an die vorhergehenden anschließen.

Genauso wird jeweils Byte 1 bis 7 für alle Blöcke einer Reihe ausgewertet und daraufhin mit der nächsten Blockzeile fortgefahren. Es ergibt sich ein Algorithmus, der vollständig durch das Struktogramm in Bild 10 beschrieben wird. Dieses Struktogramm wurde in Listing 5 in ein Basic-Programm umgesetzt. Wieder werden Sie in wesentlichen Teilen die Übereinstimmungen zum Programm zur Berechnung eines Blocks im Multicolor-Modus erkennen.

Als Ergebnis Ihrer Bemühungen dürften Sie eine Hardcopy Ihres im Computer befindlichen Bildes erhalten, das etwa so wie das Beispiel in Bild 1 aussieht.

Auf ein Wort: Basic

Nun kennen Sie zwar die Geheimnisse, die es Ihnen ermöglichen, selbst hochauflösende Grafiken auf dem Drucker auszugeben, aber spätestens wenn Sie eines der beiden Programme aus Listing 3 oder 4 eintippen und anschließend starten, wird Ihre Freude gedämpft sein, denn der Computer macht seinem Namen alle Ehre: er rechnet. Die Berechnungen nehmen aber so viel Zeit in Anspruch, daß man mitunter die Geduld verliert. Was hier also wünschenswert erscheint, ist eine Programmiersprache, die die ganze Sache etwas beschleunigt. Das naheliegendste: Benutzung von Maschinensprache. Dadurch verkürzt sich zum Beispiel die Ausgabe einer HiRes-Grafik auf 30 Sekunden (etwa durch das in der 64'er-Ausgabe 8/84 auf Seite 83 veröffentlichte Listing). Das werden Sie zu schätzen wissen, wenn Sie einmal das entsprechende Basic-Programm haben laufen lassen und dann die Maschinenroutine dagegenhalten.

Wenn Sie also Maschinensprache beherrschen und nun das Prinzip der Grafikausgabe auf dem Drucker verstehen, empfehle ich Ihnen, eine Ihren Erfordernissen angepaßte Maschinenroutine zu schreiben, um Grafik auf dem Drucker auszugeben.

Deshalb gebe ich hier dem interessierten Maschinensprachеprogrammierer Informationen über eine möglichst einfache Ansteuerung des Druckers. Dazu zunächst eine Liste aller verwendbaren Systemroutinen und Zero-Page-Adressen:

BSOUT	$FFD2	Ausgabe eines Bytes auf aktivem Kanal
OPEN	$FFC0	Eröffnen eines Files auf dem Drucker
CKOUT	$FFC9	Ausgabe auf definiertem Kanal einleiten
CLRCH	$FFCC	Ausgabe wieder auf Standard zurücksetzen
CLOSE	$FFC3	Schließen eines Files auf dem Drucker
SETPAR	$FFBA	Setzen der Parameter für OPEN
LAENGE	$B7	für die Länge eines Strings bei OPEN

Bevor Sie ein File auf dem Drucker eröffnen, ist dem Computer erst mitzuteilen, welche Filenummer, welche Primäradresse und welche Sekundäradresse für das zu öffnende File zu verwenden sind. Da insbesondere beim Öffnen keine Zeichenkette an den Drucker zu senden ist, muß zunächst einmal das Flag LAENGE auf Null gesetzt werden:

```
LDA #0
STA LAENGE
```

Daraufhin wird in den Akku die Filenummer, in das X-Register die Primäradresse des Druckers (normalerweise 4) und in das Y-Register die Sekundäradresse eines Files geladen, das dann später durch OPEN eröffnet wird. Zuerst müssen aber mit der Systemroutine SETPAR die Werte für einen OPEN-Befehl gesetzt werden, bevor die OPEN-Routine angesprungen werden darf.

```
LDA #Filenummer
LDX #Primäradresse
LDY #Sekundäradresse
JSR SETPAR
JSR OPEN
```

Damit ist ein Kanal für die Ausgabe auf dem Drucker definiert. Die eigentliche Ausgabe auf dem Drucker wird durch das nochmalige Laden der Filenummer in das X-Register und das Anspringen der Systemroutine CHKOUT eingeleitet.

```
LDX #Filenummer
JSR CHKOUT
```

Danach kann man Zeichen, die im Akku stehen, mittels der Routine BSOUT auf dem Drucker ausgeben.

```
LDA #auszugebendes Zeichen
JSR BSOUT
```

Durch diese Routine wird die Ausgabe, die mittels der Routine BSOUT normalerweise auf dem Bildschirm erfolgen würde, auf den Drucker umgeleitet. Das heißt unter anderem, daß dann während dieser Umleitung keine Zeichenausgabe auf dem Bildschirm vorgenommen werden kann. Wollen Sie die Umleitung wieder aufheben, so können Sie dies mit Hilfe von CLRCH erreichen:

```
JSR CLRCH
```

Diese Systemroutine setzt die Ausgabe auf den Bildschirm (und die Eingabe auf die Tastatur) zurück. Beachten Sie bitte, daß das Druckerfile noch nicht geschlossen ist und Sie jederzeit wieder mit der beschriebenen Syntax bei der Systemroutine CHKOUT die Ausgabe erneut umleiten können.

Das Druckerfile schließen können Sie mit der Routine CLOSE. Dazu laden Sie die Filenummer in den Akku:

```
LDA #Filenummer
JSR CLOSE
```

Danach ist das Drucker-File ordnungsgemäß geschlossen. So, damit hätten wir unsere umfangreiche Expedition in das Gebiet der Grafikumsetzung für Druckausgaben abgeschlossen. Ich hoffe, daß Sie Ihnen ein wenig Spaß gemacht und Ihr Interesse an der Problematik geweckt hat. Ich wünsche Ihnen jedenfalls viel Erfolg bei der Programmierung einer eigenen Grafikroutine, sei es nun in Basic oder einer anderen Programmiersprache.

(Frank Lonczewski/ev)

```
100 REM ****************************       <112>
110 REM *                          *       <159>
120 REM *  HIRES-BILD VERSCHIEBUNG *       <174>
130 REM *                          *       <179>
140 REM ****************************       <152>
150 DATA 165,001,072,169,000,133,251,133   <140>
160 DATA 253,169,032,133,254,120,169,052   <247>
170 DATA 133,001,162,032,160,000,177,251   <205>
180 DATA 145,253,200,208,249,230,252,230   <244>
190 DATA 254,202,208,240,104,133,001,088   <020>
200 DATA 096                               <176>
210 FOR I=4096 TO 4136:READ A:POKE I,A:PS=
    PS+A:NEXT I                            <023>
220 IF PS<>6115 THEN PRINT"PRUEFSUMMENFEHL
    ER":STOP                               <158>
230 FOR I=1 TO 7:PRINT"{CLR}GRAFIKBILDSCHI
    RM:";I                                 <230>
240 PRINT"{DOWN}STARTADRESSE:"I*8192:POKE
    252,I*32                               <253>
250 SYS 4096:GOSUB 280:PRINT"{CLR}"        <132>
260 POKE 53265,PEEK(53265)OR 32:POKE 53272
    ,PEEK(53272)OR 8:GOSUB 280             <002>
270 POKE 53265,PEEK(53265)AND 223:POKE 532
    72,PEEK(53272)AND 247:NEXT I:END       <208>
280 FOR J=1 TO 2000:NEXT J:RETURN          <108>
```

© 64'er

Listing 1. HiRes-Bildverschiebung. Bei der Eingabe bitte den Checksummer 64 in diesem Heft beachten.

```
100 REM ********************************   <238>
110 REM *                              *   <159>
120 REM * MATRIX- UMRECHNUNGSPROGRAMM *    <165>
130 REM *                              *   <179>
140 REM ********************************   <022>
150 INPUT"BASIS (X*8192) ";BASIS           <202>
160 INPUT"ZEILE (0-24){3SPACE}";ZEILE      <008>
170 INPUT"BLOCK (0-39){3SPACE}";BLOCK      <114>
180 DIM BYTE(7)                            <237>
190 FOR I=0 TO 7                           <003>
200 BYTE(I)=PEEK(BASIS+ZEILE*320+BLOCK*8+I
    )                                      <050>
210 NEXT I:PRINT:PRINT"AUSGABE DER BERECHN
    ETEN WERTE:":PRINT                     <078>
220 FOR I=0 TO 7:REM 7 BYTES VERTIKAL      <191>
230 OUTPUT=0:FOR I0=0 TO 7                 <017>
240 OUTPUT=OUTPUT-((BYTE(I0)AND 2↑(7-I))>0
    )*2↑(7-I0):NEXT I0                     <017>
250 PRINT OUTPUT:NEXT I                    <147>
```

© 64'er

Listing 2. Matrix-Umrechnungsprogramm. Bei der Eingabe bitte den Checksummer 64 in diesem Heft beachten.

```
100 REM********************************    <150>
110 REM*                              *    <159>
120 REM* HIRESBILDAUSGABE AUF DRUCKER *    <170>
130 REM*                              *    <179>
140 REM********************************    <190>
150 INPUT"BASIS (X*8192) ";BASIS:REM EINGA
    BE DER BASIS                           <074>
160 OPEN 1,4,4:REM EROEFFNEN DES DRUCKERS
    MIT LINEARKANAL =>SEKUNDAERADRESSE 4   <148>
170 PRINT#1,CHR$(27);"3";CHR$(24);:REM ZEI
    LENVORSCHUB AUF 24/216"                <091>
180 DIM BYTE(7)                            <237>
185 FOR J=0 TO 24:REM ZEILEN 1-25          <182>
```

Listing 3. HiRes-Bildausgabe auf Drucker. Bitte beachten Sie den Checksummer-Artikel.

```
186 PRINT#1,CHR$(27);"K";CHR$(64);CHR$(1);
    :REM SENDEN DES CODES FUER HIRES EIN      <160>
187 FOR K=0 TO 39:REM BLOECKE 1-39            <006>
190 FOR I=0 TO 7:REM AKTUELLER BLOCK UMREC
    HNEN                                      <150>
200 BYTE(I)=PEEK(BASIS+J*320+K*8+I)           <118>
210 NEXT I                                    <038>
220 FOR I=0 TO 7:REM 7 BYTES VERTIKAL         <191>
230 OUTPUT=0:FOR I0=0 TO 7                    <017>
240 OUTPUT=OUTPUT-((BYTE(I0)AND 2↑(7-I))>0
    )*2↑(7-I0):NEXT I0                        <017>
250 PRINT#1,CHR$(OUTPUT);:NEXT I:REM BLOCK
    AUSGEBEN, ENDE BLOCKUMRECHNUNG            <218>
260 NEXT K:REM ENDE BLOECKE EINER REIHE       <232>
270 PRINT#1,CHR$(10):REM ZEILENVORSCHUB       <018>
280 NEXT J:REM ENDE SCHLEIFE EINER REIHE      <083>
290 PRINT#1,CHR$(27)"2":REM ZEILENVORSCHUB
    WIEDER AUF 1/6" BRINGEN                   <221>
300 CLOSE 1:REM SCHLIESSEN DRUCKERKANAL       <184>
```

© 64'er

Listing 3. HiRes-Bildausgabe auf Drucker. (Schluß) Bei der Eingabe bitte den Checksummer 64 in diesem Heft beachten.

```
100 REM ******************************        <112>
110 REM *                            *        <159>
120 REM *     MULTICOLORMATRIX       *        <226>
130 REM *   UMRECHNUNGSPROGRAMM      *        <135>
140 REM *                            *        <189>
150 REM ******************************        <162>
160 DATA 0,0,0,0,0,0,0,0:REM RASTERMUSTER
    BITPAAR (00)                              <189>
170 DATA 2,2,1,1,2,2,1,1:REM RASTERMUSTER
    BITPAAR (01)                              <109>
180 DATA 2,2,0,0,1,1,0,0:REM RASTERMUSTER
    BITPAAR (10)                              <074>
190 DATA 3,3,3,3,3,3,3,3:REM RASTERMUSTER
    BITPAAR (11)                              <220>
200 DIM DRUCKMATRIX(3,7):REM DREI RASTERMU
    STER MIT JEWEILS 7 DATEN                  <128>
210 FOR I=0 TO 3                              <021>
220 FOR I0=0 TO 7                             <026>
230 READ DRUCKMATRIX(I,I0)                    <048>
240 NEXT I0,I                                 <030>
250 INPUT"BASIS (X*8192) ";BASIS              <046>
251 INPUT"ZEILE (0-24) {3SPACE}";ZEILE        <099>
252 INPUT"BLOCK (0-39) {3SPACE}";BLOCK        <196>
260 OPEN 1,4,4:REM EROEFFNEN DES DRUCKERS
    MIT LINEARKANAL =>SEKUNDAERADRESSE 4      <250>
270 PRINT#1,CHR$(27);"3";CHR$(5);:REM ZEIL
    ENVORSCHUB AUF 6/216"                     <034>
290 FOR REIHE=0 TO 7 :REM REIHEN EINES BLO
    CKS                                       <025>
300 PRINT#1,CHR$(27);"Z";CHR$(32);CHR$(0);    <033>
310 REM BIMODE 4-FACHE DICHTE EIN (MAX 192
    0 PUNKTE)                                 <045>
330 BYTE=PEEK(BASIS+ZEILE*320+BLOCK*8+REIH
    E)                                        <031>
340 FOR I=6 TO 0 STEP-2:REM 4 BITPAARE AUS
    ZUWERTEN                                  <166>
350 RASTER=-((BYTE AND 2↑(I+1))>0)*2-((BYT
    E AND 2↑I)>0)                             <169>
360 FOR I0=0 TO 7                             <168>
370 PRINT#1,CHR$(DRUCKMATRIX(RASTER,I0));:
    REM DRUCKMATRIX AUSGEBEN                  <050>
380 NEXT I0,I:REM AUSWERTUNG ENDE             <153>
400 PRINT#1,CHR$(10):REM ZEILENVORSCHUB UM
    6/216"                                    <057>
410 NEXT REIHE:REM NAECHSTE DES BLOCKS BEA
    RBEITEN                                   <107>
420 PRINT#1,CHR$(27);"2":REM ZEILENVORSCHU
    B WIEDER AUF 1/6" BRINGEN                 <198>
430 CLOSE 1                                   <187>
```

© 64'er

Listing 4. Umrechnungsprogramm für Multicolor-Matrix. Bei der Eingabe bitte den Checksummer 64 in diesem Heft beachten.

```
100 REM ******************************        <112>
110 REM *                            *        <159>
120 REM *  MULTICOLORBILDAUSGABE     *        <076>
130 REM *       AUF DRUCKER          *        <013>
140 REM *                            *        <189>
150 REM ******************************        <162>
160 DATA 0,0,0,0,0,0,0,0:REM RASTERMUSTER
    BITPAAR (00)                              <189>
170 DATA 2,2,1,1,2,2,1,1:REM RASTERMUSTER
    BITPAAR (01)                              <109>
180 DATA 2,2,0,0,1,1,0,0:REM RASTERMUSTER
    BITPAAR (10)                              <074>
190 DATA 3,3,3,3,3,3,3,3:REM RASTERMUSTER
    BITPAAR (11)                              <220>
200 DIM DRUCKMATRIX(3,7):REM DREI RASTERMU
    STER MIT JEWEILS 7 DATEN                  <128>
210 FOR I=0 TO 3                              <021>
220 FOR I0=0 TO 7                             <026>
230 READ DRUCKMATRIX(I,I0)                    <048>
240 NEXT I0,I                                 <030>
250 INPUT"BASIS (X*8192) ";BASIS              <046>
260 OPEN 1,4,4:REM EROEFFNEN DES DRUCKERS
    MIT LINEARKANAL =>SEKUNDAERADRESSE 4      <250>
270 PRINT#1,CHR$(27);"3";CHR$(5);:REM ZEIL
    ENVORSCHUB AUF 5/216"                     <161>
280 FOR ZEILE=0 TO 24:REM ZEILEN              <009>
290 FOR REIHE=0 TO 7 :REM REIHEN EINES BLO
    CKS                                       <025>
300 PRINT#1,CHR$(27);"Z";CHR$(0);CHR$(5);     <184>
310 REM BIMODE 4-FACHE DICHTE EIN (MAX 192
    0 PUNKTE)                                 <045>
320 FOR BLOCK=0 TO 39:REM FUER ALLE BLOECK
    E                                         <255>
330 BYTE=PEEK(BASIS+ZEILE*320+BLOCK*8+REIH
    E)                                        <031>
340 FOR I=6 TO 0 STEP-2:REM 4 BITPAARE AUS
    ZUWERTEN                                  <166>
350 RASTER=-((BYTE AND 2↑(I+1))>0)*2-((BYT
    E AND 2↑I)>0)                             <169>
360 FOR I0=0 TO 7                             <168>
370 PRINT#1,CHR$(DRUCKMATRIX(RASTER,I0));:
    REM DRUCKMATRIX AUSGEBEN                  <050>
380 NEXT I0,I:REM AUSWERTUNG ENDE             <153>
390 NEXT BLOCK                                <198>
400 PRINT#1,CHR$(10):REM ZEILENVORSCHUB       <148>
410 NEXT REIHE,ZEILE                          <020>
420 PRINT#1,CHR$(27);"2":REM ZEILENVORSCHU
    B WIEDER AUF 1/6" BRINGEN                 <198>
430 CLOSE 1                                   <187>
```

© 64'er

Listing 5. Multicolor-Bildausgabe auf Drucker. Bei der Eingabe bitte den Checksummer 64 in diesem Heft beachten.

Funktionen auf Tastendruck

Mit diesem Beitrag zeigen wir Ihnen, wie auf Tastendruck beliebige Funktionen oder Programme auch während eines Programmlaufs gestartet werden können. Egal ob es sich nun um ein Basic- oder Maschinenprogramm handelt.

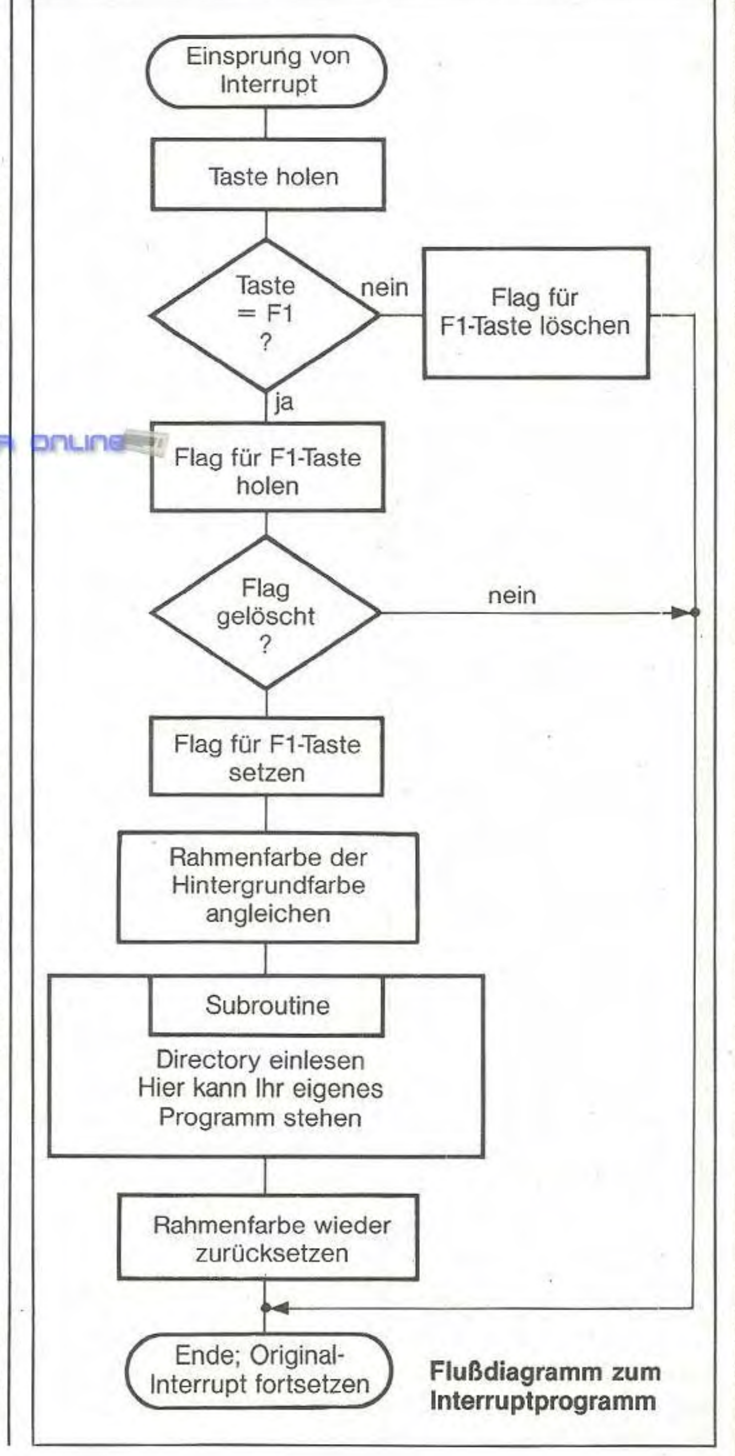
Dieser Artikel soll Ihnen das Thema »Funktionen« anhand eines Beispiels näherbringen. Das Beispielprogramm können Sie mit etwas Assemblerkenntnis auch für Ihre Programme verwenden. Vorschlag: Der F1-Taste wird eine Funktion zugeordnet. Durch Drücken der F1-Taste soll jederzeit das Directory einer Diskette am Bildschirm gezeigt werden, ohne daß ein im Speicher vorhandenes Programm gelöscht wird. Während das Directory geladen wird, soll der Bildschirmrahmen die Farbe des Hintergrundes annehmen.

Es wurden schon viele solcher Utilities veröffentlicht. Diese Programme werden aber im Regelfall durch Drücken der STOP/RESTORE-Tasten wieder abgeschaltet. Sollen diese Routinen dagegen gewappnet sein, so versagen sie spätestens nach einem Reset. Im Gegensatz zu diesen »Billigmethoden« soll aber unsere Directory-Funktion der F1-Taste auch nach einem Reset erhalten bleiben.

Um derartige Anforderungen zu realisieren, müssen einige Betriebssystemroutinen zumindest angeschnitten werden. Die Wichtigste ist die Interruptroutine des C 64. Die Interruptroutine ist ein computerinternes Programm, das sehr oft angesprochen wird. Befindet sich der Computer im »Ready«-Modus, wird die Interruptroutine beispielsweise alle 1/60 Sekunde durchlaufen. Wird der C 64 in Anspruch genommen, etwa durch Diskettenoperationen, wird dieses Programm jedoch seltener bearbeitet. Der Interrupt erfüllt viele Aufgaben. Er stellt beispielsweise die Uhrzeit (TI,TI$) nach, läßt den Cursor blinken und übernimmt die Tastaturabfrage. Das Interruptprogramm befindet sich ab Adresse $EA31 im Betriebssystem. Der Computer sucht sich diese Adresse über einen Wegweiser. Dieser Wegweiser steht bei Adresse $0314 und $0315 in der Zero-Page. Ein solcher Wegweiser wird im Fachjargon als Vektor bezeichnet. Da sich der Vektor im RAM befindet, kann dieser verändert werden. Das heißt, daß der Interrupt »umgeleitet« werden kann. Doch davon später.

Wie funktioniert eine Modulerkennung?

Eine zweite gewichtige Betriebsroutine ist die Modulerkennung. Diese benötigt man, damit die geplante Funktion der F1-Taste »resetfest« wird und zugleich immun gegen die Tastenkombination STOP/RESTORE. Die Kennungs-Routine überprüft, ob sich ein Steckmodul im Expansion-Port befindet. Und zwar an der Codefolge »cbm80« ab Adresse $8004. Diese ASCII-Codefolge entspricht den Hex-Zahlen $C3,$C2,$CD,$38 und $30. Findet der Computer diese fünf Zahlen ab Adresse $8004, werden sowohl bei einem Reset als auch beim Drücken der Restore-Taste nicht mehr die Standardroutinen angesprochen. Der C 64 springt die Adressen an, deren Vektoren in $8000/8001 und $8002/8003 stehen. Der erste Vektor ist der Resetvektor. Der Resetvektor zeigt auf die Adresse, die nach einem Reset angesprungen wird. Nach STOP/RESTORE holt sich der C 64 die Sprungadresse des NMI-Vektors aus den Speicherzellen $8002 und 8003. Ist kein Modul eingesteckt, ist der Adreßbereich ab $8000 als RAM freigegeben. Wenn man nun ab Adresse $8004 die oben genannte Codefolge ablegt, nimmt der C 64 an, daß sich ein Modul im Expansion-Port befindet. Durch diesen Trick kann der Reset- und NMI-(RESTORE-) Vektor verändert werden.



Flußdiagramm zum Interruptprogramm

Da unser Programm wegen dieses Tricks mitten im Basic-Speicher steht, muß es vor dem Überschreiben durch lange Basic-Programme oder Stringvariablen gesichert werden. Dies erreicht man, wenn man den Zeiger für den Stringspeicher $33-$34 und den Zeiger für die Speichergrenze auf $8000 setzt.

Doch nun zum Programm selbst. Nach dem Eingeben des Programmes mit Hilfe des MSE kann es mit SYS 32819 oder mit einem Reset (SYS 64738) initialisiert werden. Dabei wird der Interrupt auf die Adresse $8046 umgeleitet und das Programm vor dem Überschreiben geschützt. Solange der Interruptvektor verändert wird, muß der Interrupt gesperrt werden. Das heißt, daß während dieser Zeit die Interruptroutine nicht angesprungen werden darf. Ein Interrupt muß auch verhindert werden, wenn gerade ein Interruptprogramm stattfindet. Die Katze soll sich schließlich nicht in den eigenen Schwanz beißen. Um einen Interrupt zu sperren, gibt es einen speziellen Assemblerbefehl, den SEI (Set Interrupt Flag). Ist das IRQ-Flag (IRQ, Interrupt Request) gesetzt, läßt sich der 6502-Prozessor nicht mehr durch einen Interrupt unterbrechen, um ein anderes Programm auszuführen. Um den Interruptvektor auf unser Beispielprogramm zu richten, muß in Speicherzelle $0314 das Lowbyte ($46) und in $0315 das Highbyte ($80) unserer Programmadresse ($8046) geschrieben werden. Danach kann das Interruptflag wieder gelöscht werden. Der Assemblerbefehl dazu lautet CLI (Clear Interrupt Flag). Wird nun ein Interrupt ausgelöst (von den CIAs), springt der Prozessor zur Adresse $8046. In unserem Beispiel setzt dort die Abfrage der F1-Taste ein. Ist sie gedrückt oder nicht? Eine Antwort darauf liefert die Adresse $CB in der Zero-Page. Denn der Inhalt von $CB gibt Auskunft darüber, welche Taste zuletzt gedrückt wurde. War es die F1-Taste, steht dort eine »4«.

Den Code einer jeden Taste können Sie mit der folgenden Basic-Anweisung leicht ermitteln:

FOR I=1 TO 10000 : PRINT PEEK(203) : NEXT I

Drücken Sie danach die gewünschte Taste, erscheint die zugehörige Zahl am Bildschirm.

Das Beispielprogramm reagiert nun folgendermaßen: Ist die F1-Taste nicht gedrückt, also der Inhalt der Speicherzelle ungleich 4, wird in Speicherzelle $8009 der Wert $FF geschrieben und die Adresse $EA31 angesprungen. Wie schon erwähnt, steht ab $EA31 die normale IRQ-Routine des C 64. Warum in Speicherzelle $FF geschrieben wird, soll später erklärt werden.

Wurde die F1-Taste gedrückt, wird das Beispielprogramm bearbeitet. Dabei verändert der Bildschirm seine Rahmenfarbe und das Unterprogramm »Directory« ab Adresse $8074 wird abgearbeitet. Ab Adresse $8074 kann jedes beliebige Unterprogramm stehen. Voraussetzung ist nur, daß dieses mit RTS abgeschlossen wird. In diesem Beispiel ist es eben das Zeigen eines Diskettendirectories.

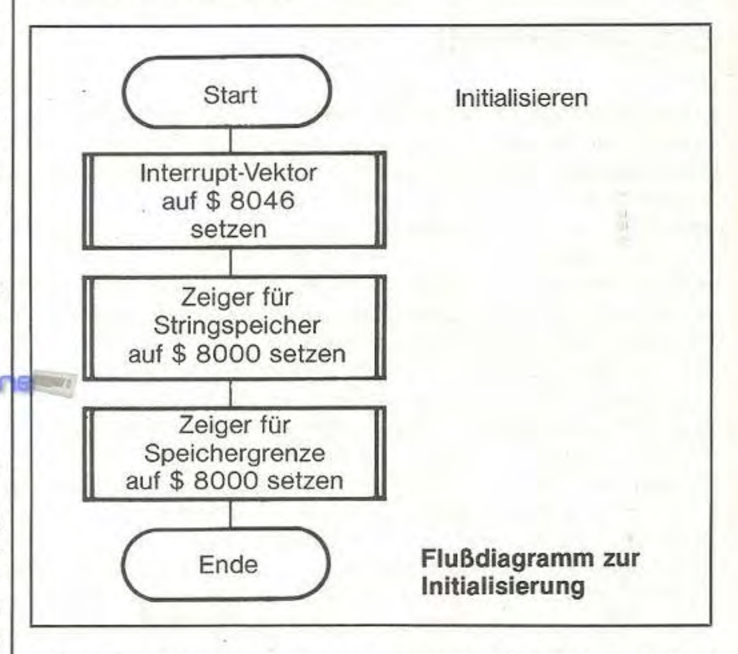
Stellen Sie sich jetzt vor, es würde kein Directory gelesen, sondern irgendeine Funktion aufgerufen, die einen sehr kleinen Zeitbedarf hat. Eventuell eine Einstellung von Bildschirmfarben, wo verfügbare Farben durch Drücken der F1-Taste gezeigt werden. Alle 1/60 Sekunde würde dann die Farbe wechseln. Unmöglich! Deshalb ist in die Tastenabfrage noch ein Trick eingebaut, der das verhindert. Dazu wird nach dem Drücken von F1 ein Flag gesetzt (Speicherzelle $8009 auf $00) und erst dann die Funktion aufgerufen. Ist beim nächsten Interrupt $8009 immer noch »0«, so heißt das, daß die F1-Taste in der Zwischenzeit noch nicht losgelassen wurde. Die Funktion soll dann nicht nochmal ausgeführt, sondern mit der originalen IRQ-Routine fortgefahren werden.

Wäre die F1-Taste in der Zwischenzeit losgelassen worden, hätte in der Speicherzelle $CB eine Änderung stattgefunden. Unsere IRQ-Routine hätte das erkannt und in $8009 den Hex-Wert $FF geschrieben.

Damit die Funktions-Belegung der F1-Taste nicht durch einen Reset zerstört werden kann, wurde das Prinzip der Modulerkennung genutzt. Der Resetvektor zeigt auf die Adresse $800A. Ab dieser Adresse wurde einfach der Anfang des Originalresets nachgebildet. Nachdem der Arbeitsspeicher neu initialisiert und alle Vektoren (auch Interruptvektor) mit ihren Standardwerten belegt wurden, springt »unser« Reset nach $8033 und initialisiert wieder die F1-Taste. Anschließend wird der Prozessor wieder auf seinen gewohnten Weg geschickt ($FCFB).

Unempfindlich gegen Reset

Der NMI-Vektor wurde auf die Adresse $801F »verbogen«. Ab dieser Adresse ist der Original-NMI nachgebaut. Bis auf den Sprung an die Adresse $FD15, der weggelassen wurde. Dort wird nämlich die Interruptadresse korrigiert und die F1-Taste abgeschaltet, was die ganzen Anstrengungen zunichte machen würde.



Flußdiagramm zur Initialisierung

Das Beispielprogramm kann natürlich beliebig erweitert oder geändert werden. Es wäre zum Beispiel möglich, die Buchstabentasten über die CTRL-Taste mit einer vierten Funktion zu belegen. Ähnlich SHIFT/CBM. Ein anderer Einsatz wäre der Aufruf einer Hardcopy. Versuchen Sie einfach mal, eine Hardcopyroutine über ????? aufzurufen, statt die Directory-Funktion. Sie müssen dazu nur den JSR $8074-Befehl ab Adresse $806A so ändern, daß die von Ihnen gewählte Routine abgearbeitet wird. Der Befehl lautet dann vielleicht JSR $C000. Haben Sie keine Angst vor Maschinensprache! Experimentieren Sie doch einfach mit diesem Programm. Eine einfache Übung: Stellen Sie mit der F1-Taste die Farben ein.

(Christian Quirin Spitzner/hm)

```
programm : directory          8000 80ef
---------------------------------------

8000 : 0a 80 1f 80 c3 c2 cd 38   1c
8008 : 30 ff 8e 16 d0 20 a3 fd   37
8010 : 20 50 fd 20 b7 e4 20 15   29
8018 : fd 20 33 80 4c fb fc 20   db
8020 : bc f6 20 e1 ff d0 09 20   86
8028 : a3 fd 20 18 e5 6c 02 a0   e0
8030 : 4c 72 fe 78 a2 46 a0 80   64
```

```
8038 : 8e 14 03 8c 15 03 58 a9  41
8040 : 80 85 34 85 38 60 a5 cb  f5
8048 : c9 04 f0 08 a9 ff 8d 09  33
8050 : 80 4c 31 ea ad 09 80 c9  59
8058 : ff d0 f6 a9 00 8d 09 80  44
8060 : ad 20 d0 48 ad 21 d0 8d  9d
8068 : 20 d0 20 74 80 68 8d 20  49
8070 : d0 4c 31 ea a9 01 20 c3  bb
8078 : ff a9 24 8d f0 03 a9 30  35
8080 : 8d f1 03 a9 01 a2 08 a0  82
8088 : 00 20 ba ff a9 02 a2 f0  5e
8090 : a0 03 20 bd ff 20 c0 ff  76
8098 : a9 40 20 90 ff a2 01 20  d5
80a0 : c6 ff 20 90 ff 20 cf ff  c0
80a8 : 20 cf ff 20 cf ff 20 cf  d1
80b0 : ff c9 00 f0 31 a9 01 85  21
80b8 : cc a2 01 20 c6 ff 20 cf  a6
80c0 : ff a8 20 cf ff 48 98 aa  0f
80c8 : 68 20 cd bd a9 20 20 d2  2d
80d0 : ff 20 cf ff c9 00 d0 08  c3
80d8 : a9 0d 20 d2 ff b8 50 cb  09
80e0 : 20 d2 ff b8 50 eb a9 01  8d
80e8 : 20 c3 ff 20 cc ff 60 00  3c
```

Listing zum Directory auf Tastendruck. Beachten Sie den MSE

```
8000  0a 80                  ; reset-vektor
8002  37 80                  ; nmi-vektor
8004  c3 c2 cd 38 30         ; cbm80 (bewirkt 'modul-start')
8009  ff
-------------------------------- reset
800a  8e 16 d0   stx $d016
800d  20 a3 fd   jsr $fda3
8010  20 50 fd   jsr $fd50   ; arbeitsspeicher initialisieren
8013  20 b7 e4   jsr $e4b7
8016  20 15 fd   jsr $fd15   ; hardware und i/o vektoren setzen/holen
8019  20 33 80   jsr $8033   ; initialisierung der f1-taste
801c  4c fb fc   jmp $fcfb   ; zurueck zum original-reset
-------------------------------- nmi
801f  20 bc f6   jsr $f6bc   ; flag fuer stop-taste setzen
8022  20 e1 ff   jsr $ffe1   ; stop-taste abfragen
8025  d0 09      bne $8030   ; nicht gedrueckt, dann $8030
8027  20 a3 fd   jsr $fda3   ; i/o initialisieren
802a  20 18 e5   jsr $e518   ; bildschirm loeschen (i/o initialisieren)
802d  6c 02 a0   jmp ($a002) ; zum basic-warmstart
8030  4c 72 fe   jmp $fe72   ; weiter im orginal nmi
-------------------------------- initialisierung der f1-taste
8033  78         sei         ; setzen des interrupt-dissable-bit
8034  a2 46      ldx #$46    ;
8036  a0 80      ldy #$80    ;
8038  8e 14 03   stx $0314   ; interrupt auf adresse
803b  8c 15 03   sty $0315   ;   $8046 setzen
803e  58         cli         ; loeschen des interrupt-dissable-bit
803f  a9 80      lda #$80    ; programm vor ueberschreiben sichern
8041  85 34      sta $34     ; zeiger fuer stringspeicher auf $8000
8043  85 38      sta $38     ; zeiger speichergrenze auf $8000
8045  60         rts
-------------------------------- erweiterung der interrupts
8046  a5 cb      lda $cb     ; aktuelle taste holen
8048  c9 04      cmp #$04    ; = f1-taste ?
804a  f0 08      beq $8054   ; ja, dann $8054
804c  a9 ff      lda #$ff    ; flag fuer
804e  8d 09 80   sta $8009   ; f1-taste loeschen
8051  4c 31 ea   jmp $ea31   ; zurueck zum interrupt
8054  ad 09 80   lda $8009   ; f1-taste schon laenger gedrueckt ?
8057  c9 ff      cmp #$ff
8059  d0 f6      bne $8051   ; ja, dann zurueck zum interrupt
805b  a9 00      lda #$00    ; setzt flag
805d  8d 09 80   sta $8009   ; fuer gedrueckte f1-taste
8060  ad 20 d0   lda $d020   ; rahmenfarbe lesen
8063  48         pha         ; farbe auf stapel legen
8064  ad 21 d0   lda $d021   ; hintergrundfarbe lesen
8067  8d 20 d0   sta $d020   ; rahmen angleichen;
806a  20 74 80   jsr $8074   ; beliebige funktion (z.b. directory)
806d  68         pla         ; rahmenfarbe von stapel holen
806e  8d 20 d0   sta $d020   ; rahmenfarbe in den originalzustand
8071  4c 31 ea   jmp $ea31   ; zurueck zum interrupt
--------------------------------directory (hier kann beliebige funktion
                                          ausgefuehrt werden.)
8074  a9 01      lda #$01
8076  20 c3 ff   jsr $ffc3   ; close 1
8079  a9 24      lda #$24
807b  8d f0 03   sta $03f0
807e  a9 30      lda #$30
8080  8d f1 03   sta $03f1
8083  a9 01      lda #$01
8085  a2 08      ldx #$08
8087  a0 00      ldy #$00
8089  20 ba ff   jsr $ffba   ; fileparameter (1,8,0) setzen
808c  a9 02      lda #$02
808e  a2 f0      ldx #$f0
8090  a0 03      ldy #$03
8092  20 bd ff   jsr $ffbd   ; filenamenparameter (2,15,3)
8095  20 c0 ff   jsr $ffc0   ; open
8098  a9 40      lda #$40
809a  20 90 ff   jsr $ff90   ; status setzen  st=64
809d  a2 01      ldx #$01
809f  20 c6 ff   jsr $ffc6   ; eingabegeraet = 1 setzen
80a2  20 90 ff   jsr $ff90   ; status setzen
80a5  20 cf ff   jsr $ffcf   ; basin (zeichen holen)
80a8  20 cf ff   jsr $ffcf   ; basin (zeichen holen)
80ab  20 cf ff   jsr $ffcf   ; basin (zeichen holen)
80ae  20 cf ff   jsr $ffcf   ; basin (zeichen holen)
80b1  c9 00      cmp #$00    ; zeichen = 0 ?
80b3  f0 31      beq $80e6   ; ja, dann ende
80b5  a9 01      lda #$01
80b7  85 cc      sta $cc     ; cursorblinken ausschalten
80b9  a2 01      ldx #$01
80bb  20 c6 ff   jsr $ffc6   ; chkin  eingabegeraet = 1 setzen
80be  20 cf ff   jsr $ffcf   ; basin (zeichen holen)
80c1  a8         tay         ; akku ins y-register
80c2  20 cf ff   jsr $ffcf   ; basin (zeichen holen)
80c5  48         pha         ; zahl auf stapel legen
80c6  98         tya         ; y->akku
80c7  aa         tax         ; akku->x
80c8  68         pla         ; zahl vom stapel holen
80c9  20 cd bd   jsr $bdcd   ; dezimalzahl ausgeben
80cc  a9 20      lda #$20
80ce  20 d2 ff   jsr $ffd2   ; leerzeichen ausgeben
80d1  20 cf ff   jsr $ffcf   ; basin (zeichen holen)
80d4  c9 00      cmp #$00    ; zeichen = 0 ?
80d6  d0 08      bne $80e0   ; nein, dann zur zeichenausgabe
80d8  a9 0d      lda #$0d
80da  20 d2 ff   jsr $ffd2   ; zeilenvorschub ausgeben
80dd  b8         clv
80de  50 cb      bvc $80ab
80e0  20 d2 ff   jsr $ffd2   ; zeichenausgabe
80e3  b8         clv
80e4  50 eb      bvc $80d1   ; naechstes zeichen holen ($80d1)
80e6  a9 01      lda #$01
80e8  20 c3 ff   jsr $ffc3   ; close 1
80eb  20 cc ff   jsr $ffcc   ; clrch
80ee  60         rts         ; zurueck zum interrupt
```

Das Quellisting zum Directory auf Tastendruck

Star SG-10 und Textomat

Haben Sie Probleme, den Star SG-10 mit Secus-Interface an Textomat anzupassen? Verzweifeln Sie nicht, hier kommt die Lösung.

Der erste und wichtigste Schritt ist die Eingabe folgender Basic-Kommandos:

OPEN 4,4,25 <Return>

PRINT#4

Die dabei verwendete Sekundäradresse »25« teilt dem Drucker mit, daß er den vollen Star-Zeichensatz fest einstellen soll. Die PRINT-Anweisung muß gegeben werden, damit der Drucker den Lock-Zustand ausführt. Um diesen Status wieder aufzuheben, muß der Drucker ausgeschaltet werden.

Als Zweites laden Sie Textomat wie gewohnt und nehmen im Menü-Punkt »Druckeranpassung« die entsprechenden Änderungen vor.

Beim Punkt »Drucker« wählen Sie die »1«. Die ASCII-Werte entnehmen Sie dem Interface-Handbuch auf Seite 14 oder dem Star-Handbuch auf den Seiten 217 bis 223. Für den Bereich »Steuerzeichen« empfehle ich folgende Befehlssequenzen:

Zeichenabstand:	di10	1b4201	(Pica)
	di12	1b4202	(Elite)
	di15	1b4203	(entspricht 17 Zeichen/inch = Schmalschrift)
Zeilenabstand:	ab1	1b4106	(1zeiliger Abstand)
	ab2	1b410c	(2zeiliger Abstand)
	ab3	1b4118	(3zeiliger Abstand)

Die anderen Steuerzeichen können Sie leicht selbst einsetzen, wobei Sie den Seitenwechsel nicht mit dem Wert 0c belegen sollten, da der Drucker dann immer zwei Seiten vorschiebt.

Sie sollten auch noch beachten, daß die DIP-Schalter am Drucker richtig eingestellt sind. Im Interface-Handbuch steht zwar, daß die Schalter 2-2, 2-3 und 2-4 auf OFF-Stellung sein sollen, ich habe aber die Erfahrung gemacht, sie besser in ON-Stellung zu lassen. Wichtigstes Argument hierfür ist das nochmalige Vorschieben um eine Zeile. Des weiteren stelle ich den 1. und 7. Schalter auf OFF.

Ich hoffe, Sie freuen sich jetzt genauso wie ich an dem tollen Schriftbild Ihres SG-10 oder SG-15 in Verbindung mit dem Textverarbeitungsprogramm Textomat.

(Klaus Croll/gk)

Zwei Quickies

Kopfstand für Sprites und ein kleiner Zeicheneditor sind zwei Einzeiler, die sich in Programme einbinden lassen.

Der folgende Einzeiler stellt ein Sprite auf den Kopf (beziehungsweise wieder auf die Füße):

Kopfstand für Sprites

FOR A=958 TO 896 STEP—3:POKE A—2,PEEK(832+L):
POKE A—1,PEEK(833+L):POKE A,PEEK (834+L):
L=L+3:NEXT

Der Einzeiler kann nur dann als solcher eingegeben werden, wenn von der Möglichkeit Gebrauch gemacht wird, die (Basic-)Kurzbefehle einzutippen (siehe Anhang D im C 64-Handbuch). Sie können jedoch auch zwei Zeilen daraus machen. Das Sprite, das verändert werden soll, muß im Bereich 13 (Adresse 832-895) abgelegt sein; im Bereich 14 (Adresse 896-958) steht nach Programmlauf das gedrehte Sprite beziehungsweise dessen Bitmuster.

A = Laufvariable für FOR...NEXT-Schleife
L = Zählvariable

Da L nicht auf Null gesetzt wird, muß vermieden werden, diese Variable anderweitig im (Gesamt-) Programm zu verwenden, da es sonst zu unliebsamen Überraschungen kommen kann. (Hans Joachim Neumann/gk)

Zeichendefinitionen

Folgendes kurze Programm wandelt ein am Bildschirm entworfenes Zeichen in Zahlen um und schreibt diese dann auf den Bildschirm. Im linken oberen Eck des Bildschirmes wird das Zeichen in einer 8 * 8-Matrix entworfen. Um einen gesetzten Punkt zu markieren, wird an die gewünschte Stelle ein Stern (»*«) geschrieben.

X(0—7)	=Zahl 0—7
T	Zähler der Reihe
Y	Zähler des Bits
1031	=linke obere Ecke des Zeichens+ 7 (1024+7)
42	Bildschirmcode für »*« — es können auch andere verwendet werden.

Damit Sie nicht mit Abkürzungen arbeiten müssen, ist der Einzeiler auf zwei Zeilen aufgeteilt worden (Zeile 100 und 110).

Starten Sie das Programm mit RUN. In der oberen linken Ecke wird ein Raster von 8*8 Punkten gezeichnet und unter dem READY erscheint GOTO 100. Mit den Cursor-Tasten (nicht die RETURN-Taste verwenden) können Sie nun ein Zeichen definieren, indem Sie anstelle eines Punktes ein »*« setzen. Wenn das Zeichen fertig ist, gehen Sie mit dem Cursor auf »GOTO 100« und drücken RETURN. Auf dem Bildschirm werden jetzt die Daten dieses Zeichens angezeigt.

Diese Daten können zum Erstellen eines eigenen Zeichensatzes (auch für Spiele) verwendet werden. Ein manuelles Errechnen dieser Werte entfällt somit.

(Wolfgang Horak/gk)

```
7 REM ZEICHENDEFINIEREN IN DER OBEREN          <107>
8 REM LINKEN BILDSCHIRMECKE                    <174>
10 PRINT"{CLR}";                               <204>
20 FOR I=1 TO 8:PRINT"........":NEXT           <239>
30 PRINT"{3DOWN}GOTO 100 "                     <253>
40 END                                         <042>
100 FOR T=0 TO 7:X(T)=0:FOR Y=7 TO 0 STEP-
    1:X(T)=X(T)-2↑Y*(PEEK(1031-Y+40*T)= 42
    ):NEXT Y                                   <055>
110 PRINT X(T);:NEXT                           <200>
© 64'er
```

3D-Darstellung in 19 Zeilen

Das Programm stellt ein frei wählbares, dreidimensionales Objekt auf dem Bildschirm dar, das durch Drücken der Funktionstasten gedreht werden kann. Simons Basic ist allerdings Voraussetzung.

Das Programm (Listing 1) wird nach Aktivierung von Simons Basic normal geladen und mit »RUN« gestartet. Danach muß die Datei angegeben werden, die die Daten für den darzustellenden Körper enthält. Bevor die Eingabe mit RETURN abgeschlossen werden kann, muß jedoch erst die Diskette mit der gewünschten Datei in das Laufwerk eingeführt werden. Sind alle Werte eingelesen, erscheint die Figur in Originalform auf dem Bildschirm. Sie kann nun durch Drücken der Funktionstasten (ohne SHIFT) beliebig gedreht werden. F1, F3 und F5 bestimmen die Rotationsachse, während F7 einen Richtungswechsel bewirkt.

Die Erzeugung von Figuren

Die Daten für Figuren müssen in Form einer sequentiellen Datei (Typ SEQ) vorliegen. Eine Figur besteht dabei aus einer begrenzten Anzahl von Geradenstücken, die duch ihre Anfangs- und Endpunkte (x-, y- und z-Werte) gekennzeichnet sind. Jedes Geradenstück wird also durch sechs Zahlen bestimmt. Alle Geradenstücke werden unmittelbar hintereinander in die Datei geschrieben. Ein Endekennzeichen oder sonstige Zusatzangaben sind nicht erforderlich.

Das Demonstrationsprogramm (Listing 2) erzeugt eine Datei mit dem Namen »PYR«, welche die Daten für eine Pyramide enthält. Durch Veränderung der DATA-Zeilen können auch andere Körper definiert werden. Die Koordinaten müssen jedoch so gewählt werden, daß sie nicht außerhalb des Bildschirmbereichs liegen und diesen auch nicht durch die Rotation zu verlassen drohen. Es sei dabei darauf hingewiesen, daß die z-Achse in die Tiefe und die y-Achse nach oben zeigt. Als Orientierungshilfe kann die Demo-Pyramide verwendet werden.

Darstellung eines Punktes

Um einen Raumpunkt auf dem Bildschirm darstellen zu können, muß er zunächst an das zweidimensionale Koordinatensystem angepaßt werden. Es wird hierzu die Fluchtpunktperspektive gewählt, die über die Strahlensätze herleitbar ist:

$x' = xf-(zf-z) * (xf-x)/zf$

$y' = yf-(zf-z) * (yf-y)/zf$

x'	horizontale Bildschirmkoordinate
y'	vertikale Bildschirmkoordinate
x,y,z	Koordinaten des Raumpunktes
xf,yf,zf	Koordinaten des Fluchtpunktes

Anwendungsmöglichkeiten

Das Programm kann beispielsweise zur Darstellung und räumlichen Drehung von Computergrafiken (dreidimensionale Funktionen, Gegenstände, Flächen etc.) verwendet werden. Die Erzeugung der Gebilde erfolgt dabei von einem unabhängigen Programm. Nach Erstellung der sequentiellen Datei kann das Ergebnis mittels des Rotationsprogramms betrachtet werden. Dazu kann es sinnvoll sein, noch einige Erweiterungen in das Programm einzubauen. Zum Beispiel

würden folgende zusätzlich eingegebenen Zeilen bei Überschreitung des Bildschirmbereiches einen Abbruch verhindern und statt dessen die betreffenden Geraden nur bis an den Rand zeichnen:

```
22    gosub 60
60    dx=x-v:dy=y-r:mm=0.01:if dx*dy<>0 then mm=dy/dx
62    if x<0 then y=y-mm*x:x=0
64    if x>320 then y=y-mm*(x-320):x=320
66    if v<0 then r=r-mm*v:v=0
68    if v>320 then r=r-mm*(v-320):v=320
70    if y<0 then x=x-y/mm:y=0
72    if y>200 then x=x-y/mm:y=200
74    if r<0 then v=v-r/mm:r=0
76    if r>200 then v=v-(r-200)/mm:r=200
78    if x<0 or x>320 or v>320 or v<0 then x=0:v=0
80    if y<0 or y>200 or r<0 or r>200 then y=0:r=0
82    return
```

Freilich nimmt ein derart ergänztes Programm beim LISTen mehr als eine Bildschirmseite ein, bietet aber bessere Anwendungsmöglichkeiten.

Ein völlig anderer Aspekt wäre die bloße Verwendung der mathematischen Formeln zum Einbau in eigene Programme (beispielsweise Spiele). Dazu kann es unter Umständen erforderlich sein, sie in Maschinensprache zu übersetzen. Da aber dies bereits mit einfachen 16-Bit-Rechenroutinen und wenigen Tabellen bewerkstelligt werden kann, dürfte das für den erfahrenen Programmierer kein allzu großes Hindernis darstellen.

Mathematisch gesehen: Das Problem der Rotation

Um einen Körper von allen nur erdenklichen Perspektiven darzustellen, muß er im Raum gedreht werden. Die dazu notwendigen Berechnungen sind allerdings recht kompliziert und auch nicht im Rahmen eines extrem kurzen Programms realisierbar, will man den Körper — das heißt alle seine Eckpunkte — um eine beliebig im Raum schwebende Achse rotieren lassen.

Verzichtet man jedoch auf eine beliebige Rotationsachse und beschränkt man sich auf den Sonderfall, daß diese zu einer der Achsen des Koordinatensystems (x-, y- oder z-Achse) parallel liegt, ergibt sich eine erhebliche Vereinfachung, die daraus resultiert, daß — bedingt durch die Parallelität — immer nur zwei Koordinatenangaben eines zu rotierenden Punktes verändert werden und somit das Problem in der Ebene betrachtet werden kann. Konstant bleibt immer diejenige Koordinatenangabe, die sich auf jene Koordinatenachse bezieht, zu welcher die Rotationsachse parallel ist.

Für die Drehung eines Punktes P um einen Punkt R ergeben sich folgende Rechenschritte:

A... Abszisse
O... Ordinate
Koordinaten von R und P : R (ar, or); P (ap,op)
Koordinate von P' : P' (ap', op')
W... Winkel, der den Grad der Rotation bestimmt
A, B, d, e, h, d', h' : siehe Bild 1
A = ABS (ATN((or-op)/(ar-ap)))
if ar-ap >O then A=-Aπ
if or-op >O then A=-A
e = SQR((d)(d)+(h)(j))
d = ap—ar; h=op-or
B = W+A; d'=COS(B)*e; h'=SIN(B)*e
ap' = ar+d'; op'=or+h'

(Der Absolutbetrag und die IF-Anweisungen sind nötig, um die durch Quotientenbildung verlorengegangene Vorzeicheninformation wiederzugewinnen.)

Diese mathematischen Ausführungen waren nur für den an den Grundlagen Interessierten gedacht. Natürlich kann man auch mathematisch unbelastet mit dem Programm experimentieren.

(Achim Vowe/ev)

Variablen-Übersicht:

x(100), y(100), z(100)	Feld für Daten einer Figur
e$	Dateiname
s, i, q	Zähler
l, m, n	Schnittpunkt der Rotationsachsen (x, y, z)
w	Rotationswinkel
xf, yf, zf	Koordinaten des Fluchtpunktes
v, r, p	Zwischenspeicher
x, y, z	Zwischenspeicher für aktuellen Figurenpunkt vor und nach der Rotation
a$, a%	Variablen für Tastatureingabe

Die Bedeutung der restlichen Variablen ist der mathematischen Beschreibung zu entnehmen. Die Bezeichnungen weichen zwar größtenteils voneinander ab, aber dennoch ist eine eindeutige Zuordnung anhand der Formeln möglich.

```
11 DIMX(100),Y(100),Z(100)
12 INPUT"FILE ";E$:OPEN1,8,4,E$+",S,R"
13 INPUT#1,X(S),Y(S),Z(S):S=S+1
14 IFST<>64THEN13
15 CLOSE1:L=122:M=105:N=53:W=.5
20 Q=0:HIRES0,1:XF=147:YF=65:ZF=399
21 FORI=1TOS/2:GOSUB30:V=X:R=Y:GOSUB30
23 LINEX,200-Y,V,200-R,1:NEXT
24 GETA$:IFA$=""THEN24
28 IFA$="■"THENW=-W:GOTO24
29 A%=ASC(A$)-132:GOTO20
30 X=X(Q):Y=Y(Q):Z=Z(Q):P=Q:Q=Q+1:IFA%=1
THENA=X:O=Y:T=L:U=M:GOSUB50:X=A:Y=O
32 IFA%=2THENA=X:O=Z:T=L:U=N:GOSUB50:X=A
:Z=O
33 IFA%=3THENA=Y:O=Z:T=M:U=N:GOSUB50:Y=A
:Z=O
40 X(P)=X:Y(P)=Y:X=XF-(ZF-Z)*(XF-X)/ZF
42 Y=YF-(ZF-Z)*(YF-Y)/ZF:Z(P)=Z:RETURN
50 E=SQR((A-T)^2+(O-U)^2):C=ABS(ATN((U-O
)/(T-A))):IFT-A>0THENC=-C-π
53 B=W+C:IFU-O>0THENC=-C:B=W+C
55 A=T+COS(B)*E:O=U+SIN(B)*E:RETURN
READY.
```

Listing 1. Programm zur 3D-Darstellung und Rotation. Nur mit Simons Basic lauffähig.

```
10 OPEN1,8,4,"PYR,S,W"
20 READD
30 IFD<>1E9THENPRINT#1,D:GOTO20
40 CLOSE1
100 DATA50,50,0,200,50,0
110 DATA200,50,0,200,50,100
120 DATA200,50,100,50,50,100
130 DATA50,50,100,50,50,0
200 DATA50,50,0,125,150,50
210 DATA200,50,0,125,150,50
220 DATA200,50,100,125,150,50
230 DATA50,50,100,125,150,50
240 DATA1E9
READY.
```

Listing 2. Dieses Programm erzeugt den Datensatz einer Pyramide für Listing 1.

Zeicheneditor für C 64

In der Kürze liegt hier die Würze. Statt seitenlanger Listings brauchen Sie nur eine einzige Bildschirmseite abzutippen, um den Zeichensatz Ihres C 64 ganz nach Belieben ändern zu können.

Das Programm »Character Change« (Listing 1) ist ein Hilfsprogramm, um den Zeichensatz des C 64 beliebig zu verändern und anschließend auf Diskette abzuspeichern. Das ganze Listing wirkt zwar auf den ersten Blick etwas unübersichtlich, aber dafür sind es halt auch bloß 24 Zeilen.

Der Vollständigkeit halber ist auch noch ein Demo-Programm dabei, das es ermöglicht, mit diesem veränderten Zeichensatz in eigenen Programmen zu arbeiten. Einsatzmöglichkeiten gibt es genug, etwa Spiele.

Ein paar Erläuterungen zum Programm:

Nach dem Starten mit »RUN«, fragt der Computer »KEY ?«. Hier kann ein beliebiges Zeichen über die Tastatur eingegeben werden. Allerdings muß es an der Stelle stehen, an der der Cursor blinkt, da aus dieser Speicherzelle des Video-RAMs der Bildschirmcode geholt wird.

Nachdem die Eingabe mit RETURN abgeschlossen wurde, erscheint die Matrix des Zeichens vergrößert auf dem Bildschirm. Dabei steht ein Sternchen für gesetztes, ein Punkt für ein nicht gesetztes Pixel.

Jetzt kann die Zeichenmatrix verändert werden, und zwar zeilenweise. Wieder durch Punkte und Sternchen kennzeichnen, wo ein Punkt gesetzt beziehungsweise nicht gesetzt werden soll. Danach RETURN drücken und die nächste Zeile ändern.

Sind alle acht Zeilen neu eingegeben, so kann man sich für »W« wie »weiter« und »S« wie »save« entscheiden. Drückt man »W«, so geht das Spielchen von vorne los, mit »S« wird der Zeichensatz auf Diskette abgespeichert. Drückt man eine andere Taste, so erfolgt ein Programmabbruch.

Das File des Zeichensatzes nennt sich »character ram«. Das Demo-Programm (Listing 2) lädt einen bereits veränderten Zeichensatz von Diskette und stellt dann alle 256 Zeichen auf dem Bildschirm dar.

So arbeitet das Programm

Noch ein paar Worte zur Arbeitsweise des Programms.

Zuerst wird von einem in DATAs abgelegten Maschinenprogramm mit Hilfe der Blockverschieberoutine des Betriebssystems der Zeichensatz zweimal in verschiedene RAM-Bereiche kopiert. Einmal nach $E000, einmal nach $9000. Dies ist erforderlich, um die Zeichenmatrix mit PEEK auslesen zu können, was im unter dem Betriebssystem liegenden RAM bekanntlich nicht geht. Die zweite Kopie ist die, aus der der Computer jetzt die Zeichen ausliest. Danach werden von Basic aus die Zeiger für den Beginn des Video-RAMs, den Zeichengenerator und die damit zusammenhängenden 16 KByte für den Video-Controller neu gesetzt. Das Video-RAM wird also zwangsweise mit verschoben, und zwar nach dezimal 49152.

Aus dem Bildschirmcode des eingegebenen Zeichens wird die Startadresse in der ersten Kopie des Zeichensatzes ermittelt und 8 Bytes ausgelesen. Aus diesen 8 Bytes werden dann die Strings für die Bildschirmdarstellung gebildet. Umgekehrt werden aus den eingegebenen Strings die Bytes für den Zeichensatz gebildet und anschließend in beide Kopien gePOKEt.

Ein zweites Maschinenprogramm ruft dann die SAVE-Routine auf und speichert die 4 KByte der ersten Kopie auf Diskette. Eine erste Kopie deshalb, um sie nach dem Laden in andere Programme nochmals lesen oder verändern zu können.

Das Demo-Programm enthält im wesentlichen eine weitere Maschinenspracheroutine, die den Zeichensatz nach $9000 lädt und dann nach $E000 verschiebt. Die Umschaltung der Zeiger geschieht auch hier von Basic aus, um dem Anwender zu überlassen, wann er den neuen Zeichensatz verwenden will.

(Marc Ritter/ev)

```
1 POKE 53280,0:POKE 53281,0:FOR I=832 TO 9
  11:READ A:POKE I,A:NEXT:SYS 832:POKE 850
  ,160                                     <127>
2 POKE 56,144:CLR:SYS 832:POKE 56576,PEEK(
  56576)AND 252:POKE 53272,8:POKE 648,192  <014>
3 INPUT"{CLR,YELLOW}KEY";Z$:AD=36864+(PEEK
  (53272)AND 2)*1024+8*PEEK(49157)         <242>
4 FOR I=0 TO 7:A$(I)="":Y=PEEK(AD+I):FOR K
  =7 TO 0 STEP-1:Z$=".":X=2↑K              <119>
5 IF(Y AND X)THEN Z$="*"                   <023>
6 A$(I)=A$(I)+Z$:NEXT:PRINT"{2SPACE}"+A$(I
  ):NEXT:PRINT"{HOME}":FOR I=0 TO 7:INPUT"
  {LIG.RED}";B$:A=0                        <224>
7 FOR K=0 TO 7:IF MID$(B$,8-K,1)="*"THEN A
  =A +2↑K                                  <137>
9 NEXT:POKE AD+I+20480,A:POKE AD+I,A:NEXT:
  PRINT"{RVSON}S{RVOFF}AVE/{RVSON}W{RVOFF}
  EITER":WAIT 203,63:GET C$                <226>
10 IF C$="S"THEN SYS 866:GOTO 3            <117>
11 DATA 120,169,51,133,1,169,,133,95,133,9
   0,133,88,169,208,133,96:IF C$="W"THEN 3 <178>
12 END:DATA 169,240,133,89,169,224,133,91,
   32,191,163,169,55,133,1,88,96,169,,133  <190>
13 DATA 251,169,144,133,252,162,8,32,186,2
   55,169,13,162,130,160,3,32,189,255,169  <239>
14 DATA 251,162,,160,160,32,216,255,96,67,
   72,65,82,65,67,84,69,82,32,82,65,77,32  <248>

© 64'er
```

Listing 1. Der Zeicheneditor. Bitte beachten Sie bei der Eingabe den Checksummer.

```
10 REM CHARACTER DEMO                      <186>
100 FOR I= 828 TO 900 :READ X:POKE I,X:NEX
    T                                      <057>
110 DATA 162, 8, 160, 1, 32, 186, 255, 169
    , 13, 162, 115, 160, 3, 32             <014>
120 DATA 189, 255, 169, 0, 32, 213, 255, 1
    20, 169, 51, 133, 1, 169, 0            <001>
130 DATA 133, 95, 133, 90, 133, 88, 169, 1
    44, 133, 96, 169, 240, 133, 89         <017>
140 DATA 169, 160, 133, 91, 32, 191, 163,
    169, 55, 133, 1, 88, 96, 67            <163>
150 DATA 72, 65, 82, 65, 67, 84, 69, 82, 3
    2, 82, 65, 77, 0, 0                    <165>
160 DATA 0, 0, 0                           <026>
200 POKE 56,144:CLR:SYS 828:POKE 56,160:CL
    R                                      <142>
205 POKE 53272,8:POKE 56576,PEEK(56576)AND
     252:POKE 648,192                      <189>
210 PRINT"{CLR,8DOWN}":FOR I=0 TO 255:POKE
     49152+I,I:POKE 55296+I,3:NEXT         <217>

© 64'er
```

Listing 2. Demo-Programm zum Zeicheneditor.

Impressum

Herausgeber: Carl-Franz von Quadt, Otmar Weber

Redaktion: Albert Absmeier, Volker Everts, Georg Klinge, Harald Meyer, Achim Hübner, Christian Rogge.
Fremdautoren:
Lonczewski, Mann, Weineck, Wippgen, Riethmüller, Zinner, Meierhofer, Eyssele, Del Pino, Schäfer, Amblank, Schmidt, Schwarz, Arndt, Klenz, Hohenberger, Ziesche, Paetzold, Hoffmann, Bertog, Kowall, Wilde, Jörgens, Büsch, Brendel, Eichholz, Sauer, Meyer, Günter, Rylander, Barcikowski, Vocke, Schneider, Spitzner, Croll, Neumann, Vowe, Ritter, Horak

Layout: Leo Eder (Ltg.)

Herstellung: Klaus Buck

Auslandsrepräsentation:
Schweiz: Markt & Technik Vertriebs AG, Kollerstr. 3, CH-6300 Zug, Tel. 042-223155, Telex: 862329
USA: M&T Publishing Inc.; 2464 Embarcadero Way, Palo Alto, CA 94303

Manuskripteinsendungen: Manuskripte und Programmlistings werden gerne von der Redaktion angenommen. Sie müssen frei sein von Rechten Dritter. Sollten sie auch an anderer Stelle zur Veröffentlichung oder gewerblichen Nutzung angeboten werden, so muß dies angegeben werden. Mit der Einsendung von Manuskripten und Listings gibt der Verfasser die Zustimmung zum Abdruck in von der Markt & Technik Verlags AG herausgegebenen Publikationen und zur Vervielfältigung der Programmlistings auf Datenträger. Honorare nach Vereinbarung. Für unverlangt eingesandte Manuskripte und Listings wird keine Haftung übernommen.

Vertriebsleitung: Hans Hörl

Anzeigenverwaltung und Disposition: Michaela Hörl

Verlagsleiter M&T-Buchverlag: Günther Frank

Druck: E. Schwend GmbH, Schmollerstr. 31, Schwäbisch Hall.
Auch Anschrift für Beihefter und Beilagen.

Preis: Das Einzelheft kostet DM 14,—

Vertrieb Handelsauflage: Inland (Groß-, Einzel- und Bahnhofsbuchhandel) sowie Österreich und Schweiz: Pegasus Buch- und Zeitschriften-Vertriebs GmbH, Hauptstätter Straße 96, 7000 Stuttgart 1, Telefon (0711) 764830

Verantwortlich:
Für redaktionellen Teil: Albert Absmeier
Für Anzeigen: Brigitte Fiebig

Vorstand: Carl-Franz von Quadt, Otmar Weber

Anschrift für Verlag, Redaktion, Vertrieb, Anzeigenverwaltung und alle Verantwortlichen:
Markt & Technik Verlag Aktiengesellschaft, Hans-Pinsel-Straße 2, 8013 Haar bei München, Telefon (089) 4613-0, Telex 5-22052

Aktionäre, die mehr als 25% des Kapitals halten: Otmar Weber, Ingenieur, München; Carl-Franz von Quadt, Betriebswirt, München. Aufsichtsrat: Dr. Robert Dissmann (Vorsitzender), Karl-Heinz Fanselow, Eduard Heilmayr